

**OPTIMASI *TRAVELLING SALESMAN PROBLEM WITH TIME WINDOWS* PADA SISTEM REKOMENDASI TUJUAN WISATA
DI KOTA BATU DENGAN METODE *EVOLUTION STRATEGIES***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Dicky Manda Putra Sidharta

NIM: 145150207111025



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

OPTIMASI *TRAVELLING SALESMAN PROBLEM WITH TIME WINDOWS* PADA
SISTEM REKOMENDASI TUJUAN WISATA DI KOTA BATU DENGAN METODE
EVOLUTION STRATEGIES

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh :
Dicky Manda Putra Sidharta
NIM: 145150207111025

Skripsi ini telah diuji dan dinyatakan lulus pada
30 Juli 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Nurul Hidayat, S.Pd, M.Sc
NIP. 19680430 200212 1 001

Edy Santoso, S.Si, M.Kom
NIP. 19740414 200312 1 004

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP. 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 30 Juli 2018

Dicky Manda Putra Sidharta
NIM. 145150207111025



KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul *Optimasi Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies*.

Dalam penulisan skripsi ini, penulis mendapatkan banyak dukungan dari banyak pihak. Atas bantuan yang telah diberikan, penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Nurul Hidayat, S.Pd, M.Sc selaku dosen pembimbing utama yang telah meluangkan waktu untuk memberikan pengarahan dan masukan kepada penulis.
2. Bapak Edy Santoso, S.Si, M.Kom selaku dosen pembimbing kedua sekaligus Wakil Dekan III Fakultas Ilmu Komputer Universitas Brawijaya, yang telah meluangkan waktu untuk memberikan pengarahan, masukan, dan saran kepada penulis.
3. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, P.hD selaku dekan Fakultas Ilmu Komputer Universitas Brawijaya.
4. Bapak Ir. Heru Nurwasito, M.Kom selaku Wakil Dekan I Fakultas Ilmu Komputer Universitas Brawijaya.
5. Bapak Suprpto, S.T, M.T selaku Wakil Dekan II Fakultas Ilmu Komputer Universitas Brawijaya.
6. Seluruh dosen Fakultas Ilmu Komputer Universitas Brawijaya yang telah mendidik dan memberikan ilmu serta wawasannya selama penulis menempuh pendidikan.
7. Seluruh civitas akademika Fakultas Ilmu Komputer Universitas Brawijaya yang telah memberikan bantuan selama penulis menempuh pendidikan.
8. Orang tua dan keluarga besar penulis yang senantiasa memberikan dukungan moral selama ini.
9. Miracle Fachrunnisa Almas yang telah menemani mulai awal semester 1 hingga saat ini, selalu memberikan bantuan, dukungan, semangat, dan motivasi yang mampu memberikan pikiran positif kepada penulis selama ini.
10. Teman-teman penulis yang memberikan dukungan kepada penulis.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih terdapat kekurangan, baik dari segi penyusunan, isi bahasan, ataupun penulisan. Oleh karena itu, apabila terdapat kritik dan saran yang membangun, penulis akan menerima sebagai bahan perbaikan. Akhir kata penulis berharap skripsi ini dapat bermanfaat bagi semua pembaca.

Malang, 30 Juli 2017

Penulis

dicky.sidharta@gmail.com



ABSTRAK

Dicky Manda Putra Sidharta, Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Wisata Di Kota Batu Dengan Metode *Evolution Strategies*.

Pembimbing : Nurul Hidayat, S.Pd, M.Sc dan Edy Santoso, S.Si, M.Kom

Kota Batu memiliki berbagai tempat wisata, baik wisata alam maupun wisata buatan. Banyaknya pilihan tempat wisata merupakan hal yang bagus bagi wisatawan. Namun di sisi yang lain, karena banyaknya pilihan tempat wisata, dapat menyebabkan wisatawan sulit untuk membagi waktu berwisata yang sesuai dengan harapan. Permasalahan yang sering dialami oleh wisatawan adalah ketika ingin mengunjungi suatu tempat wisata, namun tidak sempat berkunjung ke tempat wisata yang lainnya. Oleh sebab itu, wisatawan perlu memiliki gambaran rute perjalanan wisata agar terbentuk suatu jadwal wisata secara efektif. Pencarian rute wisata terbaik dengan biaya seminimal mungkin, namun dapat berkunjung ke berbagai tempat wisata secara optimal. Pencarian rute wisata tersebut mempertimbangkan variabel waktu dapat dikenal dengan sebutan *Travelling Salesman Problem With Time Windows* (TSP-TW). Permasalahan TSP-TW dapat dipecahkan dengan menggunakan suatu metode dalam konsep *Evolutionary Algorithm*, yaitu *Evolution Strategies* (ES). Dengan dasar hasil dari pengujian yang telah dilakukan menggunakan parameter ukuran populasi sebesar 90, jumlah *offspring* sebanyak 7 μ , dan jumlah generasi sebesar 6 menghasilkan nilai *fitness* tertinggi sebesar 0,0011223345. Hasil akhir yang diperoleh pada penelitian ini adalah rute wisata di Kota Batu yang optimal.

Kata kunci: *wisata, rute wisata, Evolution Strategies, ES, Travelling Salesman Problem With Time Windows, Kota Batu*

ABSTRACT

Dicky Manda Putra Sidharta, *Optimation Travelling Salesman Problem With Time Windows On Tourism Destination Recommendation System In Batu With Evolution Strategies Method.*

Supervisor : Nurul Hidayat, S.Pd, M.Sc and Edy Santoso, S.Si, M.Kom.

Batu has various tourist destinations, both natural and artificial tourism. Many choices of tourist destinations is a good thing for tourists. On the other side, because of the many choices of tourist destinations, can cause tourists difficult to divide travel time in accordance with expectations. The problem is often experienced by tourists is when they want to visit a tourism place, but did not have time to visit the other one. Tourists need to have an overview of the travel route in order to get an effective tour schedule. Looking for the best travel route with minimum cost, but it can visit some tourist places optimally. Looking for the best tour route considering time variables can be called Traveling Salesman Problem With Time Windows (TSP-TW). TSP-TW can be solved by using a method in the concept of Evolutionary Algorithm, Evolution Strategies (ES). Based on the results of tests that have been done using the following parameters are the number of population size is 90, the number of offspring is 7μ , and the number of generation is 6, that generate the highest fitness value of 0,0011223345. The final result obtained in this research is an optimal tourist route in the city of Batu.

Keywords: tourism, tourism route, Evolution Strategies, ES, Travelling Salesman Problem With Time Windows, City of Batu

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
<i>ABSTRACT</i>	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
DAFTAR KODE PROGRAM	xiii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	4
1.6 Sistematika Pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	6
2.1 Kajian Pustaka	6
2.2 Kota Batu.....	7
2.2.1 Pemerintahan.....	7
2.2.2 Pariwisata.....	8
2.3 <i>Travelling Salesman Problem With Time Windows (TSP-TW)</i>	8
2.4 <i>Evolution Strategies (ES)</i>	8
2.5 Siklus <i>Evolution Strategies (ES)</i>	8
2.5.1 Pembangkitan Populasi Awal	9
2.5.2 Reproduksi	9
2.5.3 Evaluasi.....	10
2.5.4 Seleksi.....	10
2.5.5 Kondisi Berhenti	10

2.6 Reciprocal Exchange Mutation	10
2.7 Elitism Selection	11
BAB 3 METODOLOGI	12
3.1 Studi Literatur	12
3.2 Pengumpulan Data	12
3.3 Perancangan	13
3.4 Implementasi	13
3.5 Pengujian dan Analisis	13
3.6 Kesimpulan dan Saran	14
BAB 4 PERANCANGAN.....	15
4.1 Deskripsi Sistem	15
4.2 Deskripsi Data	16
4.3 Proses Optimasi TSP-TW dengan <i>Evolution Strategies</i>	18
4.4 Perhitungan Manual	23
4.4.1 Pembangkitan Populasi Awal.....	23
4.4.2 Reproduksi	23
4.4.3 Evaluasi.....	24
4.4.4 Seleksi.....	25
4.5 Perancangan Antarmuka	26
4.6 Pengujian	26
4.6.1 Pengujian Ukuran Populasi	26
4.6.2 Pengujian Jumlah <i>Offspring</i>	27
4.6.3 Pengujian Jumlah Generasi	27
BAB 5 IMPLEMENTASI	29
5.1 Perangkat Keras	29
5.2 Perangkat Lunak	29
5.3 Batasan Implementasi	29
5.4 Implementasi Optimasi TSP-TW dengan <i>Evolution Strategies</i>	30
5.4.1 Pembangkitan Populasi Awal	30
5.4.2 Reproduksi	31
5.4.3 Evaluasi.....	32
5.4.4 Seleksi.....	35

5.5 Implementasi Antarmuka	36
BAB 6 PENGUJIAN DAN ANALISIS.....	37
6.1 Pengujian Ukuran Populasi	37
6.2 Pengujian Jumlah <i>Offspring</i>	39
6.3 Pengujian Jumlah Generasi.....	41
6.4 Pengujian Parameter Secara Global	43
BAB 7 PENUTUP	44
7.1 Kesimpulan.....	44
7.2 Saran	45
DAFTAR PUSTAKA.....	46
LAMPIRAN	47



DAFTAR TABEL

Tabel 4.1 Data Tempat Wisata	16
Tabel 4.2 Data Waktu Tempuh	17
Tabel 4.3 Tujuan Wisata Pilihan Pengguna	23
Tabel 4.4 Populasi Awal Hitung Manual	23
Tabel 4.5 Hasil Mutasi Induk P1 <i>Offspring</i> C1	24
Tabel 4.6 Hasil Mutasi Induk P1 <i>Offspring</i> C2	24
Tabel 4.7 Hasil Mutasi Induk P2 <i>Offspring</i> C3	24
Tabel 4.8 Hasil Mutasi Induk P2 <i>Offspring</i> C4	24
Tabel 4.9 Hasil Mutasi Induk P3 <i>Offspring</i> C5	24
Tabel 4.10 Hasil Mutasi Induk P3 <i>Offspring</i> C6	24
Tabel 4.11 Hasil Evaluasi Hitung Manual	25
Tabel 4.12 Hasil Seleksi Hitung Manual	25
Tabel 4.13 Hasil Akhir Manualisasi	25
Tabel 4.14 Skenario Pengujian Ukuran Populasi	27
Tabel 4.15 Skenario Pengujian Jumlah <i>Offspring</i>	27
Tabel 4.16 Skenario Pengujian Jumlah Generasi	28
Tabel 6.1 Parameter Pengujian	37
Tabel 6.2 Hasil Pengujian Ukuran Populasi	38
Tabel 6.3 Hasil Pengujian Jumlah <i>Offspring</i>	40
Tabel 6.4 Hasil Pengujian Jumlah Generasi	42

DAFTAR GAMBAR

Gambar 3.1 Tahapan Penelitian.....	12
Gambar 3.2 Model Perancangan Sistem.....	13
Gambar 4.1 Tahapan Perancangan.....	15
Gambar 4.2 Diagram Alir Proses Optimasi TSP-TW dengan <i>Evolution Strategies</i>	18
Gambar 4.3 Diagram Alir Pembangkitan Populasi Awal	19
Gambar 4.4 Diagram Alir Mutasi.....	20
Gambar 4.5 Diagram Alir Evaluasi.....	21
Gambar 4.6 Diagram Alir Seleksi.....	22
Gambar 4.7 Perancangan Antarmuka.....	26
Gambar 5.1 Implementasi Antarmuka.....	36
Gambar 6.1 Grafik Nilai <i>Fitness</i> Pengujian Ukuran Populasi	39
Gambar 6.2 Grafik Nilai <i>Fitness</i> Pengujian Jumlah <i>Offspring</i>	41
Gambar 6.3 Grafik Nilai <i>Fitness</i> Pengujian Jumlah Generasi	43
Gambar 6.4 Hasil Pengujian Global.....	43

DAFTAR KODE PROGRAM

Kode Program 5.1 Implementasi Pembangkitan Populasi Awal.....	30
Kode Program 5.2 Implementasi Reproduksi	32
Kode Program 5.3 Implementasi Evaluasi	34
Kode Program 5.4 Implementasi Seleksi.....	35



DAFTAR LAMPIRAN

A.1 Data Tempat Wisata	47
A.2 Data Waktu Tempuh	48



BAB 1 PENDAHULUAN

1.1 Latar belakang

Pariwisata merupakan salah satu dari sekian banyak sektor yang mempunyai pengaruh vital bagi suatu negara. Bahkan sektor pariwisata dapat menjadi tulang punggung perekonomian di suatu negara. Secara langsung maupun tidak langsung, pariwisata dapat meningkatkan perputaran perekonomian masyarakat. Berdasarkan laporan yang didapatkan dari *World Tourism Cities Federation* (WTCF) pada tahun 2018 pendapatan pariwisata dunia diperkirakan meningkat sebesar 6,7 persen dibandingkan dengan tahun sebelumnya. Dengan angka tersebut, pendapatan pariwisata dunia dapat mencapai 5,3 triliun USD. Pertumbuhan tersebut jauh lebih tinggi daripada perkiraan tingkat pertumbuhan PDB Global yang diprediksi sebesar 3,1 persen menurut World Bank (2017).

Sejalan dengan pertumbuhan pendapatan sektor pariwisata dunia, pariwisata juga memiliki kontribusi yang cukup penting di perekonomian Indonesia. Menurut Menteri Pariwisata Arief Yahya, dalam 3 tahun terakhir terjadi peningkatan secara terus menerus dari sektor pariwisata. Sektor pariwisata menyumbang devisa sebesar USD 12,225 miliar. Hal ini terjadi pada tahun 2015. Angka tersebut masih dibawah sektor Migas (USD 18,574 miliar), CPO (USD 16,427 miliar), dan batu bara (USD 14.717 miliar). Namun pada tahun 2016, pariwisata Indonesia mengalami lonjakan sumbangan devisa menjadi sebesar USD 13,568 miliar. Nominal ini menempatkan sektor pariwisata berada di urutan kedua, tepatnya setelah industri kelapa sawit atau CPO yang menyumbang sebesar USD 15,965 miliar. Pada tahun 2017, sumbangan devisa dari sektor pariwisata mengalami peningkatan hingga mencapai USD 14 miliar, namun masih tetap di urutan dua, di bawah CPO. Peningkatan tersebut tidak lepas dari meningkatnya jumlah wisatawan mancanegara yang berlibur di Indonesia serta pergerakan wisatawan lokal di daerah-daerah seluruh Indonesia. Seluruh peningkatan tersebut merupakan peranan pemerintah yang gencar dalam melakukan promosi wisata di seluruh daerah di Indonesia, peningkatan kualitas infrastruktur, perbaikan keamanan, serta pemulihan dari krisis global.

Salah satu daerah yang mempunyai potensi besar di sektor pariwisata adalah Kota Batu. Kota Batu dikenal sebagai destinasi yang cocok untuk berlibur dan beristirahat. Memiliki ciri khas hawa yang sejuk menjadi salah satu nilai lebih bagi Kota Batu. Selain itu, Kota Batu memiliki berbagai tempat wisata berupa wisata alam serta wisata hasil dari buatan. Pemerintah Kota Batu juga aktif menggandeng investor untuk menggali lebih dalam potensi-potensi wisata yang ada di Kota Batu. Maka, sudah menjadi hal yang wajar melihat banyaknya tempat-tempat wisata yang berdiri dalam beberapa tahun terakhir. Faktor-faktor tersebut cukup membuat Kota Batu menjadi salah satu tujuan wisata yang memiliki daya tarik tersendiri. Banyak wisatawan yang datang berlibur, terutama pada hari liburan dan akhir pekan.

Banyaknya pilihan tempat wisata merupakan hal yang bagus, tentu dapat memberikan pilihan untuk dapat dikunjungi oleh wisatawan. Namun di sisi yang lain, karena banyaknya pilihan tempat wisata, dapat menyebabkan wisatawan mengalami kesulitan untuk membagi waktu berwisata yang sesuai dengan harapan wisatawan. Permasalahan yang sering dialami oleh wisatawan adalah ketika ingin mengunjungi suatu tempat wisata, namun tidak sempat berkunjung ke tempat wisata yang lainnya. Permasalahan tersebut disebabkan banyak faktor, yang utama adalah waktu yang dimiliki terbatas. Dapat juga disebabkan faktor pendukung lainnya seperti terjebak di kemacetan, tentu dapat mengurangi waktu yang dimiliki. Pada umumnya, wisatawan ingin mengunjungi banyak tujuan wisata yang memiliki wahana atau kategori berbeda-beda. Oleh sebab itu, wisatawan perlu memiliki gambaran rute perjalanan wisata agar terbentuk suatu jadwal wisata secara efektif.

Pencarian rute wisata terbaik dengan biaya seminimal mungkin, namun dapat berkunjung di berbagai tempat wisata secara optimal. Pencarian rute wisata yang dikembangkan dengan beberapa tambahan variabel seperti keseluruhan waktu dalam perjalanan, waktu tempat wisata tersebut di buka dan waktu tempat wisata tersebut tutup maupun waktu yang terbaik dalam kunjungan berwisata. Metode pencarian rute tempat wisata tersebut dapat dikembangkan dengan *Travelling Salesman Problem With Time Windows* (TSP-TW). Dalam lingkup permasalahan TSP-TW dapat dipecahkan dengan suatu metode dalam konsep *Evolutionary Algorithm*, yaitu *Evolution Strategies* (ES). ES merupakan algoritme *meta-heuristic* yang hampir mirip cara kerjanya dengan *Genetic Algorithm* (GA). Namun yang membedakan antara ES dan GA adalah ES menggunakan mutasi sebagai operator reproduksi utamanya, sedangkan GA menggunakan *crossover* sebagai operator reproduksi utamanya.

Terdapat penelitian menggunakan topik yang sama, yang telah dilakukan oleh Priandani (2015) mengenai penjadwalan paket rute wisata di Pulau Bali. Metode optimasi yang digunakan pada kasus TSP ini adalah dengan Algoritma Genetika. Penelitian juga dilakukan oleh Widodo dan Mahmudy (2010) mengenai sistem yang digunakan untuk rekomendasi tempat wisata kuliner. Pada penelitian tersebut, Algoritma Genetika juga digunakan dalam menyelesaikan masalah pengoptimasianya.

Terdapat penelitian sebelumnya menggunakan metode yang sama, yaitu *Evolution Strategies* (ES) dalam penelitian yang dilakukan oleh Widodo (2017) membahas mengenai sistem rekomendasi wisata di kawasan Malang Raya dengan metode *Evolution Strategies* (ES) sebagai penyelesaiannya. Penelitian ini menghasilkan kesimpulan bahwa metode *Evolution Strategies* dapat digunakan untuk menyelesaikan studi kasus optimasi TSP-TW, dengan paramater utama yang meliputi jumlah populasi dan jumlah generasi, yang berpengaruh pada nilai *fitness* yang dihasilkan. Penelitian juga dilakukan oleh Vista (2015) mengenai optimasi dalam distribusi barang secara dua tahap dengan metode *Evolution Strategies*.

Berdasarkan penjelasan di atas, maka penulis mengusulkan metode *Evolution Strategies* (ES) untuk menyelesaikan permasalahan pada penelitian ini dengan

judul Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies*. Penulis memilih *Evolution Strategies* karena metode ini merupakan salah satu metode dalam Algoritma Evolusi yang memiliki perhitungan komputasi secara efektif. Operator-operator perhitungannya mampu mencari ruang solusi yang luas secara optimal. Selain itu, *Evolution Strategies* dapat di-hybrid dengan metode optimasi yang lainnya. Maka, penulis mengharapkan dengan adanya penelitian yang menggunakan metode *Evolution Strategies* ini dapat menghasilkan solusi jadwal rute wisata yang optimal, sehingga dapat berguna bagi para wisatawan agar dapat dengan mudah menentukan tempat wisata sebagai tempat yang akan dikunjungi.

1.2 Rumusan masalah

Berdasarkan dengan latar belakang, maka rumusan masalah yang dapat diperoleh pada penelitian ini sebagai berikut:

1. Bagaimana penerapan metode *Evolution Strategies* (ES) dalam pencarian rute perjalanan wisata terbaik di Kota Batu.
2. Bagaimana menentukan parameter metode *Evolution Strategies* (ES) yang tepat dalam pencarian rute perjalanan wisata terbaik di Kota Batu.
3. Bagaimana pengaruh parameter metode *Evolution Strategies* (ES) terhadap nilai *fitness* dari solusi yang dihasilkan dalam pencarian rute perjalanan wisata terbaik di Kota Batu.

1.3 Tujuan

Tujuan dari penelitian ini:

1. Menerapkan metode *Evolution Strategies* (ES) dalam pencarian rute perjalanan wisata terbaik di Kota Batu.
2. Menentukan parameter metode *Evolution Strategies* (ES) yang tepat dalam pencarian rute perjalanan wisata terbaik di Kota Batu.
3. Mencari pengaruh parameter metode *Evolution Strategies* (ES) terhadap nilai *fitness* dari solusi yang dihasilkan dalam pencarian rute perjalanan wisata terbaik di Kota Batu.

1.4 Manfaat

Manfaat penelitian dengan adanya penelitian ini diharapkan dapat memudahkan wisatawan dalam menentukan rute perjalanan wisata di Kota Batu agar perjalanan wisata menjadi lebih optimal.

1.5 Batasan masalah

Batasan masalah yang ditentukan pada penelitian ini adalah:

1. Rute perjalanan wisata yang dioptimasi pada metode *Evolution Strategies* (ES), menggunakan variabel waktu tempuh dan waktu operasional tempat wisata. Apabila terdapat kendala waktu mulai wisata tidak sesuai dengan waktu operasional tempat wisata, maka disebut kasus *penalty*.
2. Waktu wisata terbatas hanya pada satu hari.
3. Parameter yang digunakan adalah ukuran populasi (μ), *offspring* (λ), sigma (σ), dan jumlah generasi sebagai parameter utama. Terdapat parameter pendukung meliputi tujuan wisata, waktu mulai wisata, dan waktu selesai wisata.
4. Dalam penelitian ini data jarak tempat wisata yang digunakan merupakan data yang dibuat oleh penulis sendiri berdasarkan perkiraan waktu tempuh dari satu tempat wisata ke tempat wisata yang lain mengacu pada informasi di *Google Maps*.
5. Data waktu operasional tempat-tempat wisata yang digunakan pada penelitian ini berdasarkan pada informasi di situs Google.
6. Tempat wisata yang terdapat dalam sistem adalah tempat wisata di Kota Batu.
7. Jumlah tempat wisata yang dapat dipilih oleh pengguna harus lebih dari satu, agar proses mutasi dapat dijalankan.
8. Penelitian ini melakukan beberapa pengujian, yakni pengujian parameter meliputi pengujian ukuran populasi, pengujian jumlah *offspring*, dan pengujian jumlah generasi.
9. Rute yang dibuat oleh sistem tidak memperhitungkan pengaruh dari faktor kepadatan lalu lintas serta faktor-faktor lain yang berada di luar kendali sistem.

1.6 Sistematika Pembahasan

Pada sub bab sistematika pembahasan berisi penjelasan serta gambaran dari tujuan yang diharapkan pada penulisan penelitian ini yang terdiri dari beberapa bab sebagai berikut:

BAB 1 PENDAHULUAN

Dalam bab 1 yaitu Pendahuluan, membahas mengenai apa latar belakang yang mendasari penulis dalam melakukan penelitian. Dikemukakan pula hal-hal yang berkaitan dengan pemilihan judul penelitian, rumusan masalah penelitian, tujuan penelitian, manfaat penelitian, batasan masalah penelitian, dan sistematika pembahasan penelitian terkait *Optimasi Travelling Salesman Problem With Time*

Windows Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode Evolution Strategies.

BAB 2 LANDASAN KEPUSTAKAAN

Pada bab ini akan dibahas mencakup dasar teori sebagai acuan dalam mendukung penelitian ini yaitu mengenai Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies*.

BAB 3 METODOLOGI

Pada bab ini akan membahas tentang penjelasan metode dan tahapan penelitian serta algoritma yang digunakan dalam penelitian Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies*.

BAB 4 PERANCANGAN

Pada bab ini berisi tentang proses pada perancangan sistem. Dalam bab ini juga akan dijabarkan mengenai perancangan manualisasi, perancangan antarmuka pada sistem, serta perancangan pada evaluasi sistem.

BAB 5 IMPLEMENTASI

Dalam bab ini berisi tentang implementasi sistem. Implementasi sistem yang akan dibuat adalah berdasar pada proses perancangan yang telah dilakukan sebelumnya.

BAB 6 PENGUJIAN DAN ANALISIS

Bab ini berisi tentang hasil pengujian yang dilakukan serta memberikan analisisnya.

BAB 7 PENUTUP

Bab ini membahas tentang kesimpulan yang diperoleh berdasarkan hasil pengujian, serta memberikan saran yang bertujuan untuk membangun penelitian selanjutnya agar memperoleh hasil yang lebih baik.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Pada kajian pustaka dalam penelitian ini membahas tentang perbandingan penelitian yang hendak dilakukan dengan penelitian yang telah dilakukan sebelumnya. Penelitian yang telah dilakukan sebelumnya dapat meliputi penelitian dengan objek penelitian yang sama, namun dengan metode yang berbeda. Ada pula penelitian dengan metode yang sama, namun objek penelitiannya berbeda. Penelitian-penelitian tersebut akan dibandingkan dengan cara membandingkan objek, parameter, masukan, metode yang digunakan, serta keluaran yang dihasilkan.

Penelitian yang dilakukan oleh Priandani (2015) mengenai penggunaan Algoritma Genetika sebagai optimasi penjadwalan paket rute wisata di Pulau Bali. Pada penelitian tersebut menjelaskan tentang pengoptimasian penjadwalan rute wisata di Pulau Bali. Menggunakan parameter *PopSize*, *Crossover Rate*, *Mutation Rate*, dan banyaknya generasi yang akan dibentuk. Penelitian ini menyimpulkan bahwa didapatkan hasil yang optimal dengan menggunakan metode seleksi elitism. Diperoleh hasil rata-rata *fitness* sebesar 0,000476833. 1750 generasi didapatkan sebagai jumlah generasi paling optimal. Dari hasil pengujian jumlah populasi, sebanyak 100 populasi diperoleh sebagai jumlah populasi paling baik. Untuk nilai terbaik pada *Crossover Rate* adalah sebesar 0,05 dan nilai *Mutation Rate* yang paling baik adalah 0,35.

Pada penelitian yang dilakukan oleh Widodo (2017) mengenai optimasi TSP dengan kasus rekomendasi wisata di Malang Raya menggunakan metode *Evolution Strategies* dalam menyelesaikan masalah optimasinya. Penelitian tersebut mengangkat topik yang hampir sama dengan penelitian ini, namun penelitian ini lebih spesifik membahas tentang optimasi rute perjalanan wisata di Kota Batu. Penelitian yang dilakukan oleh Agus Wahyu Widodo menggunakan masukan data meliputi waktu berangkat wisata, waktu selesai wisata, kategori wisata, jumlah destinasi, jumlah generasi serta jumlah populasi. Kemudian menghasilkan keluaran yang menampilkan rekomendasi perjalanan wisata. Penelitian ini menghasilkan kesimpulan bahwa metode *Evolution Strategies* (ES) dapat digunakan guna menuntaskan permasalahan optimasi dalam kasus TSP-TW. Pada penelitian ini parameter utama yang digunakan meliputi jumlah populasi serta jumlah generasi, yang berpengaruh pada nilai *fitness* yang dihasilkan.

Kemudian terdapat penelitian yang telah dikembangkan oleh Widodo dan Mahmudy (2010) mengenai sistem rekomendasi pada wisata kuliner. Penerapan Algoritma Genetika digunakan pada penelitian tersebut. Dalam penelitian ini diperoleh hasil bahwa rekomendasi wisata kuliner mampu merubah model dari fungsi tujuan pada TSP normal menjadi fungsi TSP dengan time-window (TSP-TW). Model tersebut melakukan pertimbangan pada ketepatan waktu kunjungan. Aplikasi ini mampu memperoleh hasil jadwal kunjungan wisata beserta rutenya

dengan akurat dan cepat. Dibuktikan bahwa GA dapat menuntaskan permasalahan dalam 40 iterasi dengan waktu <10 detik.

Ada juga penelitian yang dilakukan oleh Vista (2015). Pada penelitian ini dibahas mengenai distribusi barang secara dua tahap dengan metode optimasi algoritma genetika. Penelitian tersebut memiliki parameter penentuan optimasi, yaitu kebutuhan dari para agen dan sub agen, kapasitas produsen, serta biaya pendistribusian tanpa memperhitungkan jadwal dari produksi, serta jumlah kendaraan. Penelitian ini menggunakan masukan, yang pertama dari parameter ES yaitu μ , λ , range nilai sigma, dan jumlah generasi. Kemudian yang kedua dari parameter masalah. Pada parameter ini ditunjukkan dalam bentuk antara lain banyaknya produsen, agen, dan sub agen, kapasitas produsen, kebutuhan agen dan sub agen, serta biaya distribusi dari produsen ke agen, maupun agen ke sub agen. Jalur distribusi optimal, nilai *fitness* dari solusi yang optimal serta total biaya distribusi merupakan keluaran dari sistem. Dari hasil pengujian, parameter yang menghasilkan *fitness* hamper mendekati ukuran optimal, yaitu ukuran populasi sebesar 100, ukuran *offspring* (λ) sebesar 5μ . Pada hasil perbandingan mutasi diperoleh 20:30:50, dengan rincian segmen 1: segmen 2: segmen 3 dengan jumlah generasi 50. Nilai *fitness* sebesar 0,22441651705566 diperoleh berdasarkan penggunaan parameter tersebut.

Kemudian penelitian yang dilakukan oleh Munawaroh (2015) dengan judul Optimasi Distribusi Pupuk Menggunakan *Evolution Strategies*. Dari penelitian, dapat diketahui hasil dari pengujian parameter yang telah dilakukan, diperoleh parameter terbaik, yaitu $\mu=80$, $\lambda = 5$. Jumlah generasi yang diperoleh sebesar 40 generasi. Segmen mutasi, dengan komposisi segmen mutasi diperoleh (30%, 20%, 50%).

2.2 Kota Batu

Kota Batu merupakan salah satu kota yang berada di dataran tinggi dengan hawa sejuk pegunungan. Terletak di Provinsi Jawa Timur, Indonesia. Berjarak 90 km dari arah barat daya kota Surabaya. Jarak dari kota Malang adalah 15 km dari sebelah barat laut Kota Malang. Wilayah Kota Batu terletak di ketinggian 700-1.700 meter di atas permukaan laut. Suhu udara rata-rata di Kawasan ini mencapai 12 hingga 19 derajat Celcius. Dahulu kala, kota batu menjadi bagian dari Kabupaten Malang, kemudian ditetapkan menjadi kota otonom pada tanggal 17 Oktober 2001.

Kota Batu dikenal sebagai salah satu kota wisata di Indonesia. Hal ini dikarenakan keindahan alamnya yang luar biasa. Bahkan zaman dahulu bangsa Belanda membandingkan keindahan alam negara Swiss dengan Kota Batu.

2.2.1 Pemerintahan

Kota Batu merupakan kotamadya yang dipimpin oleh seorang wali kota dan wakil wali kota. Sejak 2007, Wali Kota Batu dan Wakil Wali Kota Batu dipilih secara langsung oleh masyarakat melalui pilkada. Sebelum dilakukan pemilihan umum oleh masyarakat, wali kota dan wakil wali kota Batu dipilih oleh anggota DPRD

kota. Wali Kota Batu dan Wakil Wali Kota Batu saat ini adalah Dewanti Rumpoko dan Punjul Santoso.

2.2.2 Pariwisata

Pariwisata yang berada Kota Batu adalah salah satu yang terbesar di Indonesia. Jumlah kunjungan wisatawannya juga yang paling banyak setara dengan Yogyakarta dan Bali. Obyek wisata di Kota Batu beragam, dari retail, sejarah, pendidikan, hingga wisata alam. Wisata-wisata tersebut meliputi:

1. Air terjun : Coban Putri, Coban Rais, Coban Talun.
2. Pemandian : Selecta, Cangar, Songgoriti.
3. Agrowisata : Arboretum, Kusuma Agrowisata.
4. Perkemahan : Gunung Panderman, Taman Hutan Raya R. Soerjo.
5. Keluarga : Jawa Timur Park 1, Museum Angkut, BNS. Eco Green Park, Predator Fun Park, Dino Park, Kusuma Waterpark, Batu Secret Zoo.
6. Kuliner : BTC, Pasar Sore.

2.3 Travelling Salesman Problem With Time Windows (TSP-TW)

Pencarian rute terpendek dan optimal termasuk dalam *Travelling Salesman Problem* (TSP). Dalam perkembangannya terdapat model TSP yang lebih kompleks. Model TSP yang lebih rumit atau kompleks disebut dengan *Travelling Salesman Problem With Time Windows* (TSP-TW). TSP-TW memiliki tambahan variabel waktu dalam penentuan rute yang optimal.

2.4 Evolution Strategies (ES)

Evolution Strategies (ES) merupakan suatu metode yang hampir mirip dengan *Genetic Algorithm* (GA). Ingo Rechenberg dan Hans Paul Schwefel merupakan orang yang pertama kali mengembangkan ES pada tahun 1960-an. Menurut Beyer dan Schwefel (2000) kegunaan dari ES adalah untuk menemukan hingga mendekati suatu solusi optimum dari suatu studi kasus (permasalahan).

Dalam ES digunakan notasi, yaitu λ (*lambda*), μ (*miu*), $\cdot \mu$ (*miu*) diartikan sebagai ukuran populasi yang didapatkan melalui proses pengacakan susunan gen-gen pada kromosom sejumlah ukuran populasi yang telah ditentukan. Sedangkan λ (*lambda*) dapat diartikan sebagai banyaknya *offspring* yang mampu dihasilkan pada proses reproduksi. Untuk menentukan jumlah *offspring*, menggunakan Persamaan 2.1.

$$\lambda = \mu * C \quad (2.1)$$

2.5 Siklus Evolution Strategies (ES)

Terdapat empat siklus di dalam ES, menurut Mahmudy (2013) yaitu:

1. (μ, λ)

Siklus (μ, λ) tidak perlu menggunakan proses rekombinasi. Sedangkan proses seleksinya hanya menggunakan individu *offspring* tanpa individu induk.

2. $(\mu/r, \lambda)$

Siklus $(\mu/r, \lambda)$ menyerupai siklus (μ, λ) . Namun, proses rekombinasi digunakan dalam siklus $(\mu/r, \lambda)$ ini. Proses seleksinya sama dengan siklus (μ, λ) hanya menggunakan individu *offspring* tanpa melibatkan individu induk.

3. $(\mu + \lambda)$

Siklus $(\mu + \lambda)$ tidak perlu menggunakan proses rekombinasi. Selain itu proses seleksi melibatkan individu *offspring* dan individu induk.

4. $(\mu/r + \lambda)$

Pada Siklus $(\mu/r + \lambda)$ memiliki persamaan yang tipis dengan siklus $(\mu + \lambda)$. Hanya dalam perbedaan antar keduanya siklus $(\mu/r + \lambda)$ menggunakan proses rekombinasi. Untuk proses seleksinya individu *offspring* dan individu induk akan dilibatkan.

Pada penelitian ini, peneliti menggunakan ES dengan siklus $(\mu + \lambda)$. Langkah-langkah penyelesaian menggunakan ES, yaitu pembangkitan populasi awal, reproduksi menggunakan mutasi, evaluasi dengan menghitung nilai *fitness*, dan seleksi menggunakan *Elitism Selection*.

2.5.1 Pembangkitan Populasi Awal

Di dalam populasi, kromosom tersusun dari sejumlah gen yang merepresentasikan variabel-variabel solusi. Pada metode *Evolution Strategies*, representasi kromosom disertai dengan fungsi *fitness* yang menyatakan kebaikan dari solusi, dan *sigma* yang menyatakan level mutasi.

Ciri utama *Evolution Strategies* adalah penggunaan vektor bilangan pecahan sebagai representasi kromosomnya. Namun dengan adanya perkembangan, ES juga dapat menyelesaikan permasalahan kombinatorial yang menggunakan representasi permutasi. Pada tahap ini akan dibangkitkan populasi awal atau disebut μ (*miu*) yang berisi individu awal serta nilai *sigma* secara acak.

2.5.2 Reproduksi

Pada ES untuk menentukan jumlah *offspring* atau disebut λ (*lambda*) menggunakan Persamaan 2.1.

C dapat dinyatakan sebagai sebuah konstanta yang mendefinisikan jumlah *offspring* dari setiap induk. Metode mutasi yang digunakan adalah *Reciprocal Exchange Mutation* yaitu dengan cara memilih 2 gen secara random (acak) kemudian 2 gen yang pada akhirnya terpilih akan ditukarkan. Metode *Reciprocal Exchange Mutation* dapat dijelaskan pada Gambar 2.1.

		XP1 ↓				XP2 ↓	
P1	1	3	2	5	6	4	
C1	1	4	2	5	6	3	

Gambar 2.1 Reciprocal Exchange Mutation

Jika terdapat paling sedikit 1/5 atau 20% hasil mutasi yang menghasilkan individu yang lebih baik dari induknya maka nilai sigma (σ) dinaikkan. Untuk menurunkan Nilai σ yaitu dengan mengalikan nilai σ dengan 0,8. Sedangkan untuk menaikkan dengan mengalikan nilai σ dengan 1,1 (Mahmudy, 2013).

2.5.3 Evaluasi

Proses dari perhitungan nilai *fitness* dari individu induk dan individu *offspring* dinamakan proses evaluasi. Setelah proses mutasi dilakukan maka akan dilanjutkan pada tahap evaluasi.

2.5.4 Seleksi

Proses guna mendapatkan individu dengan nilai *fitness* tinggi yang sesuai dengan ukuran pada populasi atau disebut juga μ (miu) dinamakan proses seleksi. *Elitism selection* adalah teknik seleksi yang biasa digunakan pada ES. *Elitism selection* adalah seleksi dengan cara mengurutkan individu-individu dari *fitness* dengan nilai paling tinggi ke nilai terendah. Setelah individu tersebut diurutkan maka akan dipilih individu terbaik. Hal ini disesuaikan dengan jumlah awal individu dalam populasi untuk menjadi induk pada tahapan generasi selanjutnya.

2.5.5 Kondisi Berhenti

Kondisi berhenti dilakukan setelah proses seleksi dijalankan serta mampu menghasilkan induk baru yang berada di dalam populasi. Setelah kriteria memenuhi maka akan dilakukan pengecekan. Menurut Beyer dan Schwefel (2002) kondisi berhenti yang biasa digunakan yaitu:

1. Ketika suatu batas maksimum iterasi atau generasi telah berhasil dicapai.
2. Ketika batas waktu pemrosesan telah terlewati.
3. Ketika terjadi konvergensi dini. Artinya, nilai *fitness* tidak mengalami perubahan yang signifikan (nilai yang dihasilkan *stuck*) atau bahkan nilai *fitness* sama sekali tidak mengalami perubahan.

2.6 Reciprocal Exchange Mutation

Proses mutasi *Reciprocal Exchange Mutation* dengan memilih dua posisi (*Exchange Point / XP*) secara acak di kromosom induk kemudian menukarkan nilai pada posisi tersebut (Mahmudy, 2013).

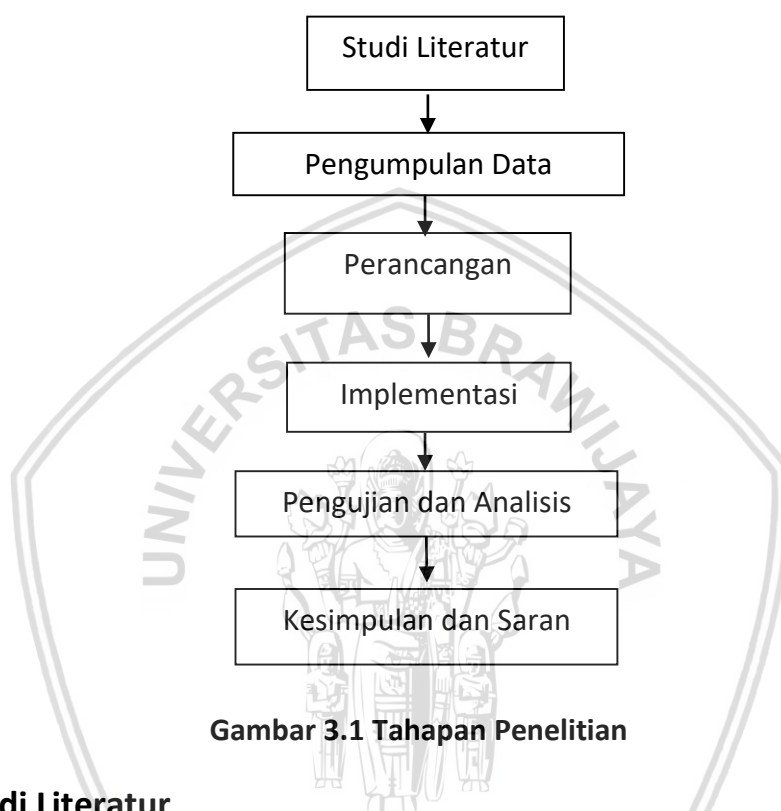
2.7 Elitism Selection

Metode seleksi *elitism* bekerja dengan mengumpulkan semua individu dalam populasi induk dan *offspring* dalam satu penampungan. *popSize* individu terbaik dalam penampungan ini akan lolos untuk masuk dalam generasi selanjutnya (Mahmudy, 2013). Metode seleksi ini menjamin individu yang terbaik akan selalu lolos. Salah satu kelemahan dari *elitism selection* adalah tidak memberikan kesempatan kepada individu dengan nilai *fitness* rendah untuk bereproduksi. Dalam beberapa kasus, solusi optimum justru bisa dicapai dari hasil reproduksi individu dengan nilai *fitness* rendah.



BAB 3 METODOLOGI

Pada bab ini akan dibahas mengenai langkah-langkah penelitian Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies* yang secara umum berdasarkan pada Gambar 3.1.



Gambar 3.1 Tahapan Penelitian

3.1 Studi Literatur

Pada penelitian yang akan dilakukan dibutuhkan studi literatur untuk menunjang penelitian ini. Studi literatur berguna untuk menyusun dasar teori pada bab 2. Dasar teori disusun berdasarkan sumber yang diperoleh dari jurnal, artikel, buku, dan penelitian-penelitian terkait. Berikut merupakan dasar teori yang dibutuhkan untuk mendukung penulisan penelitian:

1. Kajian Pustaka.
2. Teori *Evolutionary Algorithm* / Algoritma Evolusi.
3. Teori *Evolution Strategies*.

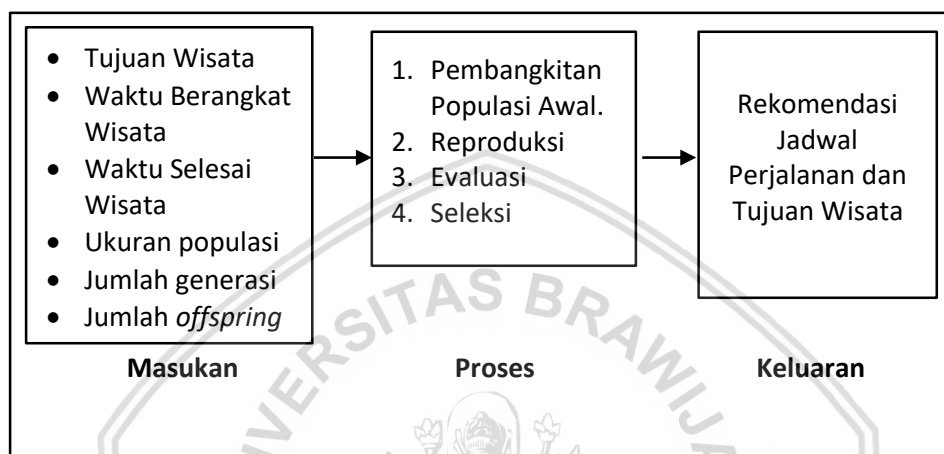
3.2 Pengumpulan Data

Pengumpulan data yang ada pada penelitian ini dilakukan dengan menggunakan teknik Data Sekunder, yaitu data telah dikumpulkan oleh pihak lain. Sebagai pembahasan, data yang digunakan adalah data perkiraan waktu tempuh dari sebuah tempat wisata menuju tempat wisata selanjutnya (tempat wisata

lain). Dengan menggunakan bantuan *Google Maps*, data tersebut dapat diperoleh. Selain itu, data waktu operasional tempat-tempat wisata di Kota Batu dapat diperoleh dengan mengakses Google.

3.3 Perancangan

Pada tahap perancangan, menjelaskan tentang proses-proses dalam sistem yang akan dijalankan menggunakan ES. Perancangan dibuat untuk memudahkan tahapan implementasi. Model perancangan dapat dilihat pada Gambar 3.2.



Gambar 3.2 Model Perancangan Sistem

3.4 Implementasi

Kemudian pada langkah ini, penelitian dilanjutkan dengan mengimplementasikan dasar teori serta perhitungan manual dari metode *Evolution Strategies (ES)* dalam Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies*.

3.5 Pengujian dan Analisis

Setelah dilakukan langkah implementasi, maka akan dilakukan pengujian terhadap program Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies*. Pengujian yang dilakukan meliputi pengujian parameter. . Parameter utama dan parameter pendukung merupakan 2 jenis parameter yang digunakan. Parameter utama meliputi ukuran populasi, jumlah generasi, dan jumlah *offspring*. Sedangkan parameter pendukung meliputi tujuan wisata, waktu berangkat wisata, dan waktu selesai wisata. Tujuan utama dari tahap pengujian pada penelitian ini adalah untuk mengetahui nilai parameter utama yang optimal, sehingga dapat dilakukan pencarian rute yang efisien.

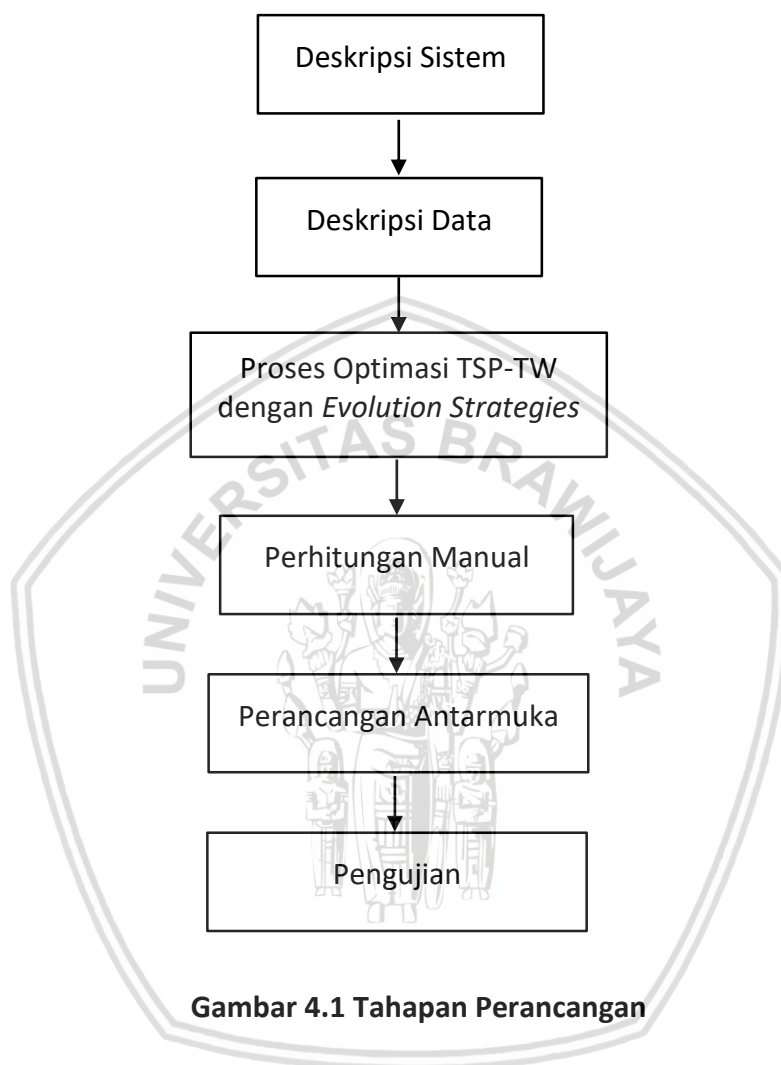
3.6 Kesimpulan dan Saran

Pada langkah terakhir setelah melakukan pengujian, maka dapat diperoleh kesimpulan berdasarkan hasil penelitian selama ini. Kesimpulan dapat menjawab perumusan masalah yang telah dirumuskan. Selain kesimpulan, langkah ini juga memberikan saran. Saran berguna bagi penelitian selanjutnya agar penelitian selanjutnya mendapat hasil yang lebih baik.



BAB 4 PERANCANGAN

Tahapan perancangan sistem pada penelitian ini terdiri dari beberapa langkah. Gambar 4.1 menunjukkan tahapan perancangan sistem.



Gambar 4.1 Tahapan Perancangan

4.1 Deskripsi Sistem

Sistem yang dibangun dalam penelitian ini adalah sistem yang mampu merekomendasikan tujuan wisata di Kota Batu beserta jadwal wisatanya dengan menerapkan sebuah metode Algoritma Evolusi yaitu *Evolution Strategies*. Pada proses menjalankannya, pengguna akan memilih tempat-tempat wisata yang akan dikunjungi, menentukan waktu keberangkatan sekaligus waktu selesai berwisata, dan jumlah generasi. Selanjutnya sistem akan mengolah masukan dari pengguna menggunakan cara kerja metode *Evolution Strategies*, yang pada hasil akhirnya akan menampilkan rekomendasi rute tempat wisata dan jadwal wisatanya yang paling optimal.

4.2 Deskripsi Data

Data dalam penelitian ini menggunakan data tempat-tempat wisata di Kota Batu, waktu tempuh dari setiap tempat wisata ke tempat wisata lainnya, jam buka dan jam tutup tempat wisata. Data tersebut diambil dari beberapa referensi di internet. Sedangkan data waktu tempuh diperoleh dari Google Maps dengan satuan menit. Berikut adalah data tempat wisata terdapat pada Tabel 4.1 dan data waktu tempuh terdapat pada Tabel 4.2

Tabel 4.1 Data Tempat Wisata

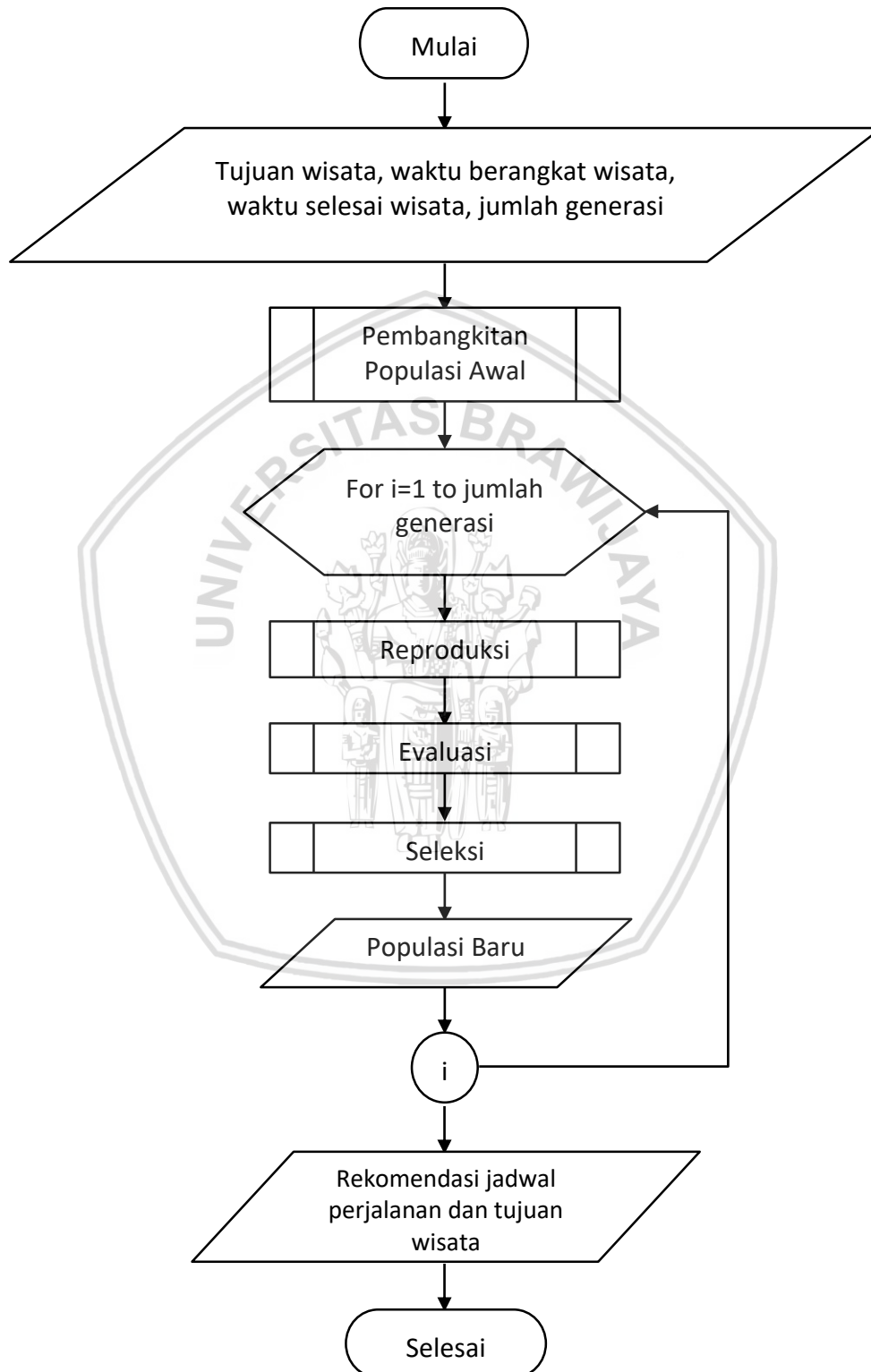
No	Nama Lokasi	Waktu	
		Buka	Tutup
1	Jawa Timur Park 1	08.30	16.30
2	Jawa Timur Park 2	09.00	18.00
3	Jawa Timur Park 3	11.00	18.00
4	Batu Secret Zoo	09.30	17.30
5	Eco Green Park	09.00	17.00
6	Batu Night Spectacular	15.00	00.00
7	Museum Angkut	12.00	20.00
8	Predator Fun Park	08.30	16.30
9	Selecta	06.00	17.00
10	Batu Wonderland	09.00	17.00
11	Kusuma Agrowisata	07.00	17.00
12	Alun-Alun Kota Wisata Batu	00.00	23.59
13	Wisata Paralayang Gunung Banyak	00.00	23.59
14	Kaliwatu Rafting	08.00	17.00
15	Coban Rais	00.00	16.00
16	Coban Talun	00.00	17.00
17	Coban Putri	00.00	16.00

Tabel 4.2 Data Waktu Tempuh

ID Tempat Wisata	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	0	6	11	6	5	7	4	16	25	5	7	8	32	12	19	30	15
2	6	0	12	1	3	6	7	16	26	6	10	11	35	14	17	31	13
3	11	12	0	10	11	9	11	8	27	7	14	10	36	15	20	32	13
4	6	1	10	0	1	4	5	13	24	4	8	7	33	13	16	30	12
5	5	3	11	1	0	6	7	15	26	6	10	9	35	15	17	31	13
6	7	6	9	4	6	0	8	9	28	7	11	11	36	16	14	33	9
7	4	7	11	5	7	8	0	17	20	5	3	5	28	8	16	26	12
8	16	16	8	13	15	9	17	0	37	16	21	20	45	25	13	42	7
9	25	26	27	24	26	28	20	37	0	24	26	18	35	17	40	15	36
10	5	6	7	4	6	7	5	16	24	0	8	5	31	10	19	27	15
11	7	10	14	8	10	11	3	21	26	8	0	9	31	12	14	30	10
12	8	11	10	7	9	11	5	20	18	5	9	0	26	5	22	23	17
13	32	35	36	33	35	36	28	45	35	31	31	26	0	26	41	44	37
14	12	14	15	13	15	16	8	25	17	10	12	5	26	0	27	22	23
15	19	17	20	16	17	14	16	13	40	19	14	22	41	27	0	43	6
16	30	31	32	30	31	33	26	42	15	27	30	23	44	22	43	0	47
17	15	13	13	12	13	9	12	7	36	15	10	17	37	24	6	47	0

4.3 Proses Optimasi TSP-TW dengan *Evolution Strategies*

Diagram alir yang menunjukkan tahapan-tahapan dari metode *Evolution Strategies* ditunjukkan pada Gambar 4.2.

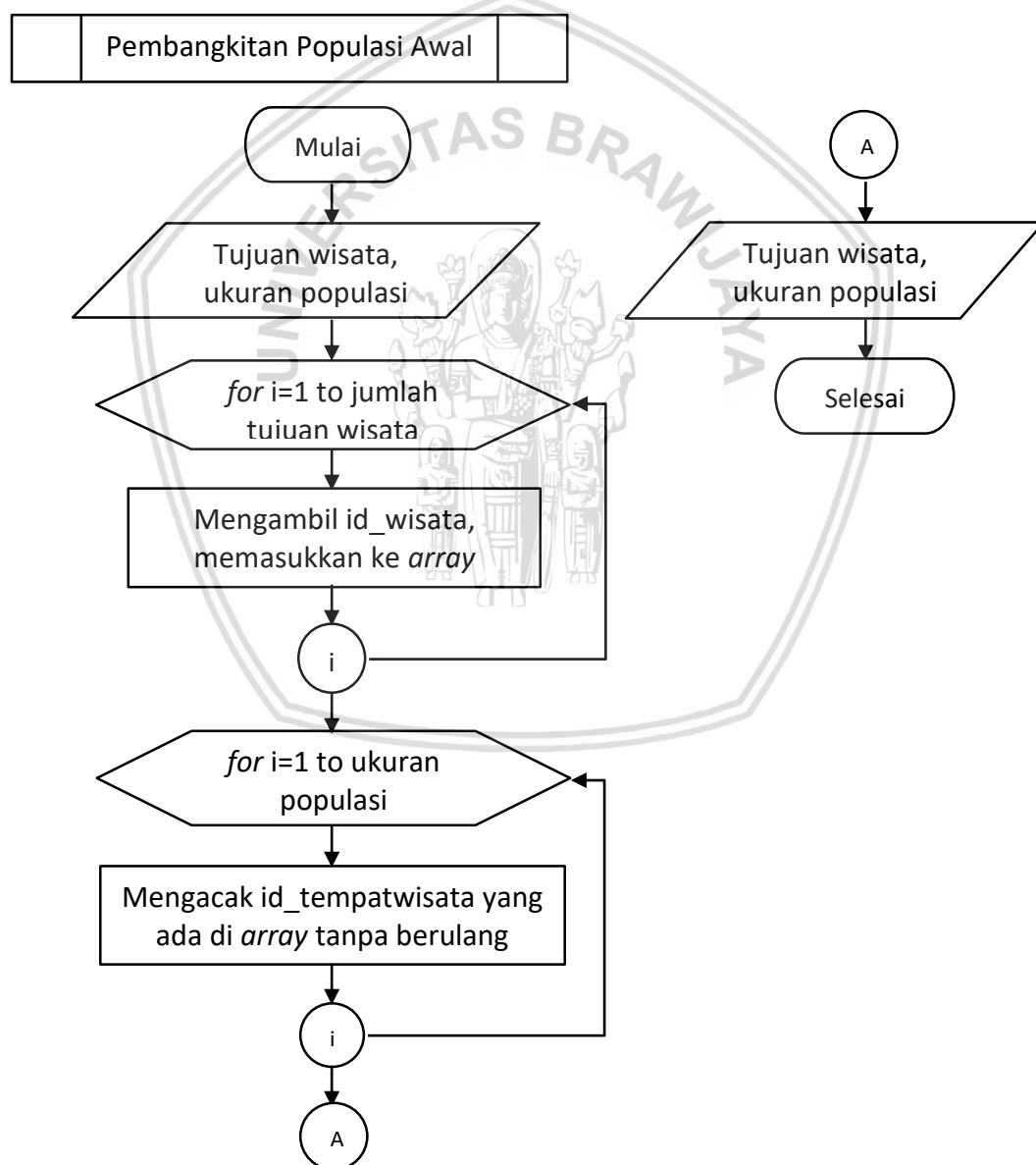


Gambar 4.2 Diagram Alir Proses Optimasi TSP-TW dengan *Evolution Strategies*

Penulis menggunakan siklus ES ($\mu + \lambda$) pada penelitian ini, dimana proses optimasi tanpa menggunakan rekombinasi. Berdasarkan diagram alir di Gambar 4.2 dapat dijelaskan secara lebih rinci tahapan-tahapan dalam proses pencarian rute tujuan wisata sebagai berikut:

1. Pembangkitan Populasi Awal

Mahmudy (2013) menyatakan bahwa pembangkitan populasi awal adalah proses membentuk kumpulan individu secara *random* (acak) yang memiliki susunan gen atau kromosom. Gen atau kromosom merupakan representasi dari solusi dari permasalahan yang hendak diselesaikan. Representasi kromosom yang digunakan pada penelitian ini adalah representasi permutasi. Tiap *string* kromosomnya mewakili indeks tempat wisata. Diagram alir dari proses pembangkitan populasi awal dapat dilihat pada Gambar 4.3.

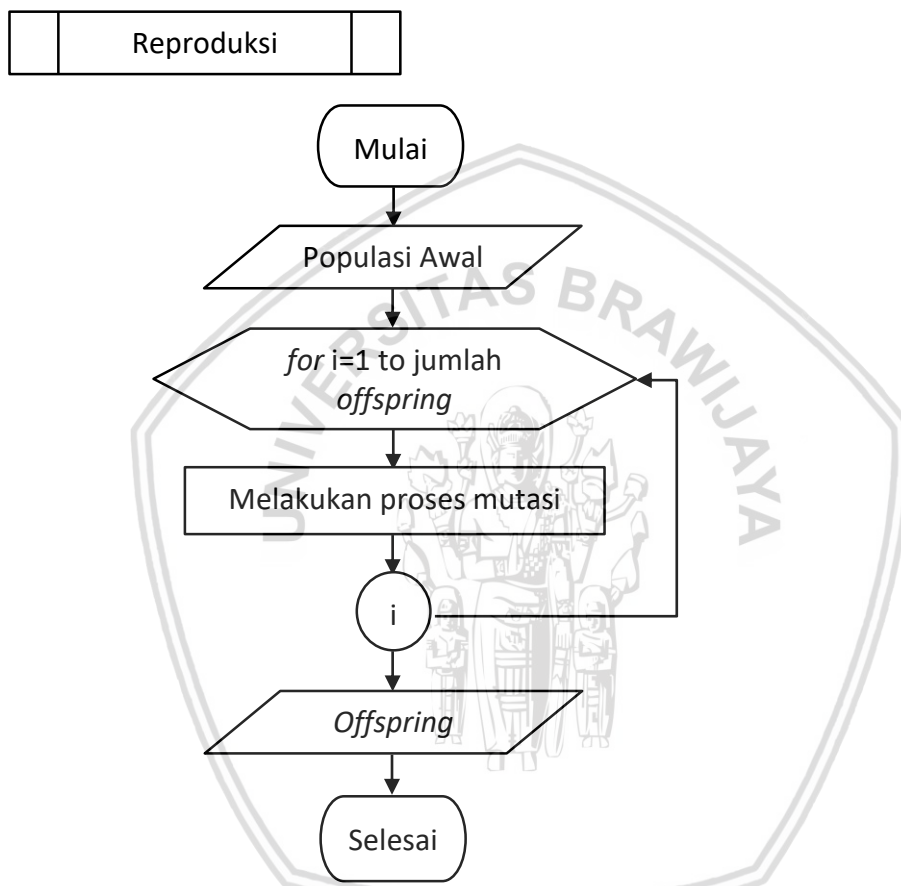


Gambar 4.3 Diagram Alir Pembangkitan Populasi Awal

Tahapan-tahapan pembangkitan populasi awal diawali dengan memasukkan tujuan wisata dan ukuran populasi. Selanjutnya menghitung jumlah tujuan wisata yang kemudian indeks tempat wisatanya disimpan di dalam *array*. Kemudian indeks tempat wisata tersebut diacak tanpa berulang sebanyak ukuran populasi yang telah ditentukan.

2. Reproduksi

Metode *Reciprocal Exchange Mutation* digunakan untuk proses reproduksi mutasi yang dapat dilihat pada Gambar 4.4.



Gambar 4.4 Diagram Alir Mutasi

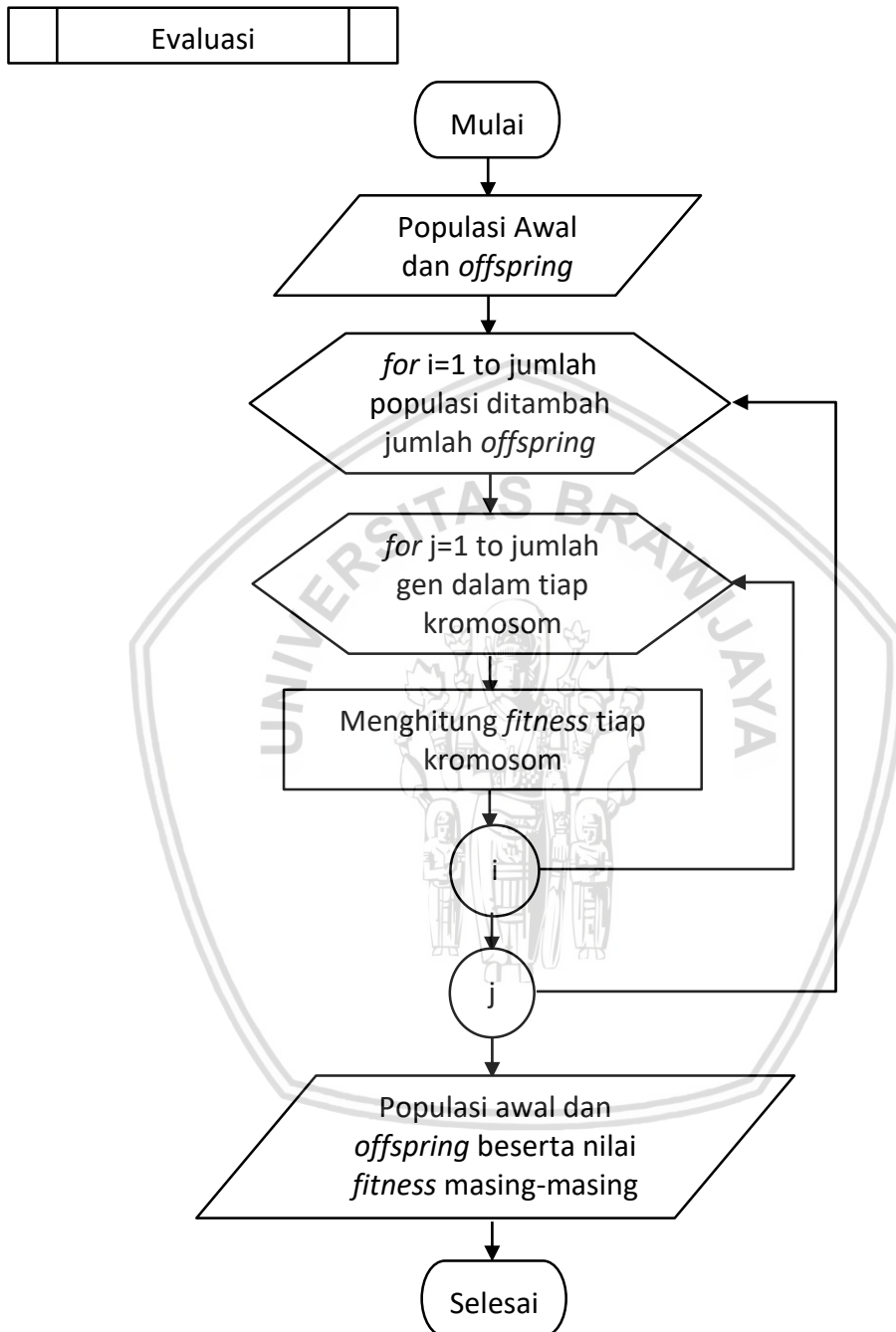
Proses mutasi dilakukan untuk membentuk beberapa *offspring* (λ) dengan menentukan jumlahnya menggunakan Persamaan 4.1.

$$\lambda = \mu * C \quad (4.1)$$

Misal, nilai C ditentukan 2. Maka setiap individu induk akan menghasilkan 2 *offspring*. Selain itu, proses mutasi juga dipengaruhi oleh nilai sigma (σ) yang melekat di setiap individu. Nilai tersebut menjelaskan setiap individu dapat melakukan proses mutasi sebanyak n kali untuk menghasilkan suatu *offspring*.

3. Evaluasi

Proses evaluasi adalah proses menghitung nilai dari *fitness* individu di populasi awal dan individu *offspring* yang dapat dilihat pada Gambar 4.5.



Gambar 4.5 Diagram Alir Evaluasi

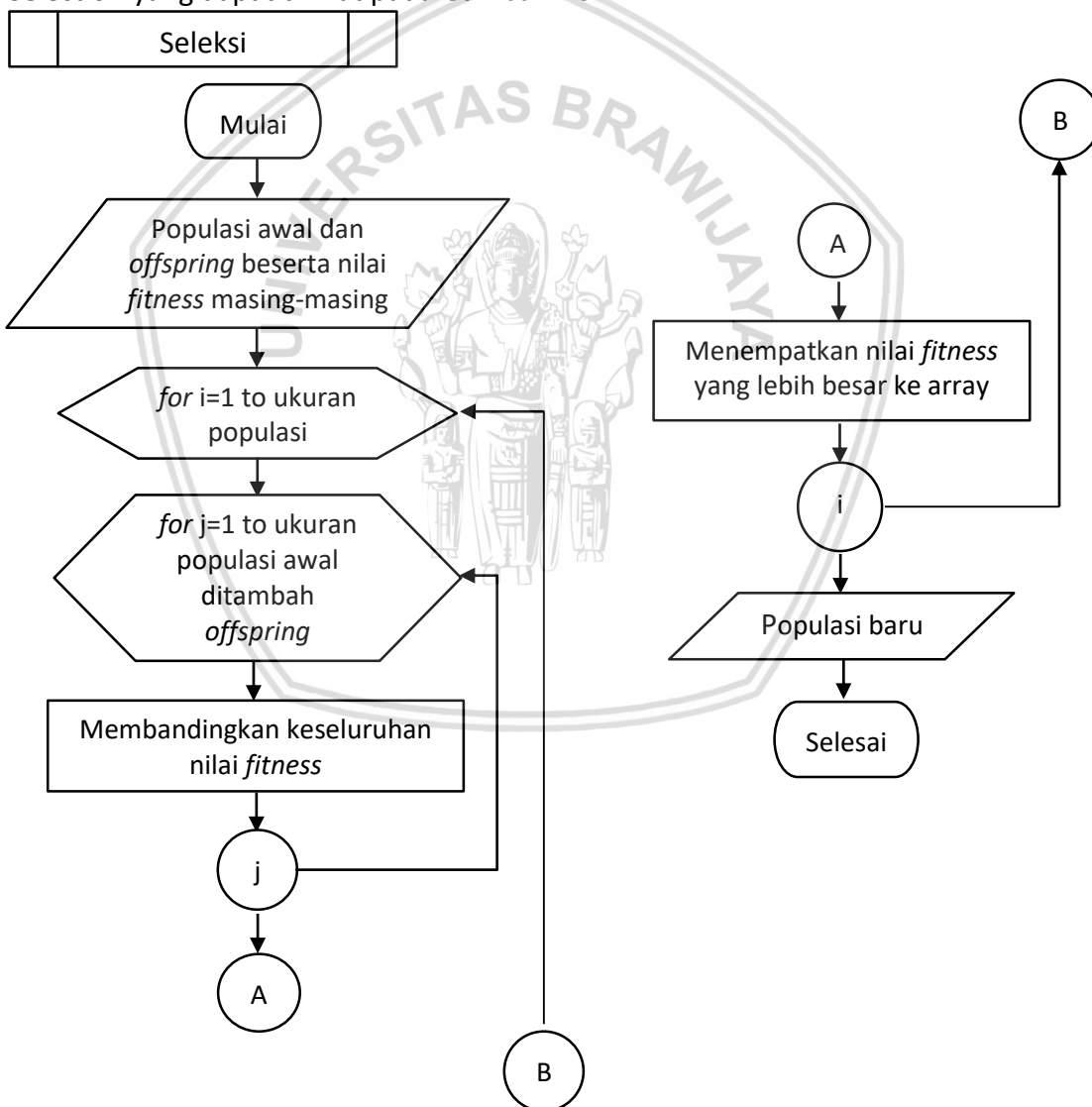
Nilai *fitness* setiap individu di populasi awal maupun *offspring* merupakan acuan dalam menentukan kromosom terbaik. Hasil dari nilai total penalty dan waktu tempuh digunakan untuk perhitungan nilai *fitness*. Nilai total waktu tempuh adalah waktu perjalanan wisata antar tempat wisata ditambah dengan waktu kunjungan di suatu tempat wisata tersebut selama 120 menit. Nilai penalti adalah

nilai yang menggambarkan kondisi apabila dalam proses optimasi terjadinya kendala. Kondisi yang pertama, apabila jam datang di suatu tempat wisata kurang dari jam buka tempat wisata, maka nilai *fitness* akan terkena penalti yaitu penambahan nilai 100. Sedangkan kondisi yang kedua, apabila jam datang di suatu tempat wisata melebihi jam tutup tempat wisata, maka nilai *fitness* ditambah 85. Jika jam selesai wisata melebihi jam tutup tempat wisata, maka terkena penalti sebesar 50. Kemudian nilai *fitness* dapat dihitung menggunakan rumus yang dapat dilihat pada Persamaan 4.2.

$$Fitness = \frac{1}{(Total Waktu Tempuh + Penalti)} \quad (4.2)$$

4. Seleksi

Penelitian ini menggunakan metode seleksi yang disebut dengan *Elitism Selection* yang dapat dilihat pada Gambar 4.6.



Gambar 4.6 Diagram Alir Seleksi

5. Populasi Baru

Populasi baru berisi individu-individu baru yang telah melewati proses seleksi antara populasi awal dan *offspring* hasil proses reproduksi. Populasi baru merupakan solusi yang dihasilkan.

4.4 Perhitungan Manual

Perhitungan manual metode *Evolution Strategies* menggunakan siklus $\mu + \lambda$. Pada tahap awal pengguna memilih tujuan tempat wisata. Setelah itu pengguna mengatur jam mulai wisata dan jam selesai wisata. Ukuran populasi ditentukan sebanyak 5, dan jumlah generasi sebanyak 2.

4.4.1 Pembangkitan Populasi Awal

Kromosom yang terbentuk berdasarkan dengan jumlah tujuan wisata yang ditentukan oleh pengguna. Misal tujuan wisata yang terpilih seperti pada Tabel 4.3 dan populasi awal yang terbentuk seperti pada Tabel 4.4.

Tabel 4.3 Tujuan Wisata Pilihan Pengguna

No	Nama Lokasi	Waktu	
		Buka	Tutup
1	Jawa Timur Park 1	08.30	16.30
4	Batu Secret Zoo	09.30	17.30
6	Batu Night Spectacular	15.00	00.00
7	Museum Angkut	12.00	20.00
9	Selecta	06.00	17.00

Tabel 4.4 Populasi Awal Hitung Manual

Individu	Kromosom					σ
P1	4	1	7	6	9	0,0575
P2	7	4	9	6	1	1,2399
P3	7	4	9	1	6	1,6768

4.4.2 Reproduksi

Dilanjutkan dengan proses reproduksi yang menggunakan mutasi. Mutasi yang digunakan adalah metode *Reciprocal Exchange Mutation*. Kemudian ditentukan nilai $\lambda = 2 \times \mu = 6$. Maka, proses mutasi ini akan membuat setiap individu menghasilkan 2 *offspring*. Proses mutasi dapat dilihat pada Tabel 4.5 hingga Tabel 4.10.

Tabel 4.5 Hasil Mutasi Induk P1 *Offspring* C1

P1	4	1	7	6	9
Proses 1	1	4	7	6	9
C1	1	4	7	6	9

Tabel 4.6 Hasil Mutasi Induk P1 *Offspring* C2

P1	4	1	7	6	9
Proses 1	7	1	4	6	9
C2	7	1	4	6	9

Tabel 4.7 Hasil Mutasi Induk P2 *Offspring* C3

P2	7	4	9	6	1
Proses 1	7	6	9	4	1
C3	7	6	9	4	1

Tabel 4.8 Hasil Mutasi Induk P2 *Offspring* C4

P2	7	4	9	6	1
Proses 1	7	4	9	1	6
C4	7	4	9	1	6

Tabel 4.9 Hasil Mutasi Induk P3 *Offspring* C5

P3	7	4	9	1	6
Proses 1	1	4	9	7	6
Proses 2	1	4	7	9	6
C5	1	4	7	9	6

Tabel 4.10 Hasil Mutasi Induk P3 *Offspring* C6

P3	7	4	9	1	6
Proses 1	7	1	9	4	6
Proses 2	7	1	6	4	9
C6	7	1	6	4	9

4.4.3 Evaluasi

Setelah melakukan proses reproduksi, akan dilanjutkan dengan proses evaluasi, yaitu menghitung nilai *fitness* semua individu di populasi awal maupun *offspring*. Hasil dari proses evaluasi pada penelitian ini dapat dilihat pada Tabel 4.11.

Tabel 4.11 Hasil Evaluasi Hitung Manual

Individu	Krosomom					σ	<i>Fitness</i>
P1	4	1	7	6	9	0,0575	0,00101
P2	7	4	9	6	1	1,2399	0,00111
P3	7	4	9	1	6	1,6768	0,00123
C1	1	4	7	6	9	0,0632	0,00113
C2	7	1	4	6	9	0,0632	0,00102
C3	7	6	9	4	1	0,9919	0,00104
C4	7	4	9	1	6	1,3639	0,00123
C5	1	4	7	9	6	1,8445	0,00141
C6	7	1	6	4	9	1,3414	0,00102

4.4.4 Seleksi

Pada penelitian ini proses seleksi menggunakan metode *Elitism Selection*. Dengan metode ini proses seleksi dengan bertahap melakukan pengurutan individu dari yang memiliki nilai *fitness* terbesar ke yang terkecil. Kemudian individu yang berjumlah sebanyak populasi akan diambil sebagai hasil akhir yang akan dilanjutkan ke generasi berikutnya. Hasil seleksi dapat dilihat pada Tabel 4.12.

Tabel 4.12 Hasil Seleksi Hitung Manual

Individu	Individu Awal	Krosomom					σ	<i>Fitness</i>
P1	C5	1	4	7	9	6	1,8445	0,00141
P2	P3	7	4	9	1	6	1,6768	0,00123
P3	C4	7	4	9	1	6	1,3639	0,00123

Setelah hasil seleksi diperoleh dan diketahui bahwa individu P1 dengan kromosom 1-4-7-9-6 adalah solusi yang terbaik. Maka proses terakhir yang harus dilakukan adalah mengembalikan angka indeks yang ada di gen tiap individu menjadi nama tempat wisata yang sesuai. Hasil akhirnya dapat dilihat pada Tabel 4.13.

Tabel 4.13 Hasil Akhir Manualisasi

Jam Mulai	Jam Selesai	Lokasi
08.30	10.30	Jawa Timur Park 1
10.36	12.36	Batu Secret Zoo
12.41	14.41	Museum Angkut
15.01	17.01	Selecta
17.29	19.29	Batu Night Spectacular

4.5 Perancangan Antarmuka

Antarmuka dalam penelitian ini hanya menggunakan satu halaman agar aplikasi terlihat lebih simpel. Selain itu untuk meminimalkan kebingungan dari pengguna. Dalam antarmuka, pengguna diwajibkan untuk memberi tanda centang atau *checkbox* pada tempat wisata yang akan dikunjungi, menentukan jam mulai wisata dan jam selesai wisata, serta ukuran populasi dan jumlah generasi. Untuk lebih jelasnya, perancangan antarmuka dapat dilihat pada Gambar 4.7.

Gambar 4.7 Perancangan Antarmuka

4.6 Pengujian

Pada tahap pengujian ini, akan dilakukan pengujian guna menguji kinerja metode *Evolution Strategies* yang telah diimplementasikan. Pengujian yang dilakukan adalah pengujian parameter dan pengujian waktu komputasi. Ukuran populasi, jumlah *offspring*, dan jumlah generasi digunakan sebagai bagian dari pengujian parameter.

4.6.1 Pengujian Ukuran Populasi

Skenario pengujian dari ukuran populasi akan dilakukan 10 kali. Masing-masing pengujian akan dilakukan sebanyak 10 percobaan. Ukuran populasi akan ditentukan sebanyak kelipatan 10 kali yang dimulai dari 10 populasi. Skenario pengujian dapat dilihat pada Tabel 4.14.

Tabel 4.14 Skenario Pengujian Ukuran Populasi

Ukuran Populasi	Nilai <i>Fitness</i>										Rata- rata <i>fitness</i>
	Percobaan										
	1	2	3	4	5	6	7	8	9	10	
10											
20											
30											
40											
50											
60											
70											
80											
90											
100											

4.6.2 Pengujian Jumlah *Offspring*

Skenario pengujian jumlah *offspring* akan dilakukan sebanyak 10 kali dengan nilai *offspring* 1 μ sampai 10 μ . Setiap pengujian akan dilakukan sebanyak 10 kali percobaan. Hal ini bertujuan untuk mengetahui nilai dari rata-rata *fitness* yang dapat diperoleh. Skenario pengujian jumlah *offspring* dapat dilihat pada Tabel 4.15

Tabel 4.15 Skenario Pengujian Jumlah *Offspring*

Jumlah Offspring	Nilai <i>Fitness</i>										Rata- rata <i>fitness</i>
	Percobaan										
	1	2	3	4	5	6	7	8	9	10	
1μ											
2μ											
3μ											
4μ											
5μ											
6μ											
7μ											
8μ											
9μ											
10μ											

4.6.3 Pengujian Jumlah Generasi

Pada tahap pengujian jumlah generasi akan dilakukan percobaan sebanyak 10 kali. Percobaan dimulai dari 1 generasi sampai 10 generasi. Masing-masing dari pengujian akan dikerjakan hingga 10 kali percobaan. Hasil dari pengujian ini diperoleh nilai rata-rata *fitness* berdasar kan perubahan jumlah generasi. Skenario pengujian jumlah generasi dapat dilihat pada Tabel 4.16

Tabel 4.16 Skenario Pengujian Jumlah Generasi

Jumlah Generasi	Nilai <i>Fitness</i>										Rata- rata <i>fitness</i>
	Percobaan										
	1	2	3	4	5	6	7	8	9	10	
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											



BAB 5 IMPLEMENTASI

Implementasi pada penelitian ini adalah tahap untuk membangun program Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies* yang telah dirancang berdasarkan pada tahap perancangan sebelumnya. Pada bab ini akan dijelaskan tentang perangkat apa saja yang digunakan, batasan implementasi, dan implementasi metode pada sistem yang dibangun.

5.1 Perangkat Keras

Pada tahap pembuatan sistem yang dibangun, terdapat perangkat keras (*hardware*) yang digunakan adalah PC (*Personal Computer*). Berikut spesifikasi dari PC yang digunakan yaitu:

1. Processor Intel(R) Core(TM) i5-6400 CPU @2.70GHz (4 CPUs),~2.7GHz.
2. Solid State Disk 110 GB.
3. Harddisk 1 TB.
4. RAM 8192 MB.

5.2 Perangkat Lunak

Perangkat lunak yang digunakan guna pembuatan sistem yang dibangun di antaranya adalah:

1. Sistem Operasi (*Operating System*) yang digunakan pada penelitian ini yaitu Windows 10 Pro 64-bit.
2. Dalam penyusunan kode program menggunakan *NetBeans* IDE 8.0.1.

5.3 Batasan Implementasi

Terdapat beberapa batasan dalam tahap implementasi sistem pada penelitian ini diantaranya adalah:

1. Dalam melakukan Implementasi sistem yang digunakan pada penelitian ini menggunakan bahasa pemrograman Java.
2. Data tempat wisata beserta waktu operasional, data jarak antar tempat wisata, dan aturan-aturan terkait dengan nilai penalti disimpan di file .txt.
3. Tampilan sistem ini berupa GUI (*Graphical User Interface*).
4. Sistem dapat menerima masukan tujuan wisata dengan jumlah pilihan sesuai keinginan pengguna.
5. Waktu mulai wisata dan selesai wisata terbatas pada satu hari.

5.4 Implementasi Optimasi TSP-TW dengan *Evolution Strategies*

Pada implementasi Optimasi TSP-TW dengan *Evolution Strategies* ini menjelaskan terkait kode program yang digunakan untuk membangun sistem sesuai dengan tahap perancangan pada bab sebelumnya.

5.4.1 Pembangkitan Populasi Awal

Pembangkitan populasi awal adalah suatu proses dalam pembentukan kromosom awal yang berisi sejumlah gen. Tujuan-tujuan wisata dan nilai *sigma* direpresentasikan oleh gen-gen ini. Jumlah dari gen berkesesuaian dengan jumlah tujuan wisata pilihan pengguna. Pada proses ini tujuan wisata pilihan pengguna akan diambil nilai indeksinya. Setelah jumlah wisata atau jumlah gen diketahui, maka akan dilakukan pengacakan susunan gen yang berarti indeks tempat wisata. Nilai *sigma* akan diacak dengan rentang nilai 0 hingga 2. Implementasi kode program pembangkitan populasi awal dapat dilihat pada Kode Program 5.1.

No.	Kode Program
1	public void inisialisasiKromosom() {
2	Random rand = new Random();
3	for (int i = 0; i < this.p.length; i++) {
4	ArrayList<Integer> pengecualianRandom = new
5	ArrayList<>();
6	for (int j = 0; j < this.p[i].length; j++) {
7	if (j < this.jmlDestinasi) {
8	int temp;
9	if (pengecualianRandom.isEmpty()) {
10	temp = 0 + (int) (Math.random() * ((this.jmlDestinasi - 1) - 0) + 1));
11	} else {
12	temp = randomDenganPengecualian(rand, 0, (this.jmlDestinasi - 1), pengecualianRandom);
13	}
14	pengecualianRandom.add(temp);
15	Collections.sort(pengecualianRandom);
16	this.p[i][j] = this.pilihanDestinasi[temp];
17	} else if (j == this.jmlDestinasi) {
18	this.p[i][j] = this.BATAS_SIGMA_BAWAH + (this.BATAS_SIGMA_ATAS - this.BATAS_SIGMA_BAWAH) * rand.nextDouble();
19	} else if (j == this.jmlDestinasi + 1) {
20	this.p[i][j] = 99;
21	} else {
22	this.p[i][j] = -1;
23	}
24	pengecualianRandom.clear();
25	}
26	}

Kode Program 5.1 Implementasi Pembangkitan Populasi Awal

Penjelasan Kode Program 5.1 yang merupakan implementasi pembangkitan populasi awal adalah sebagai berikut:

1. Baris 1 merupakan inialisasi *method* bernama inialisasiKromosom().
2. Baris 2 merupakan inialisasi fungsi *Random* dari Java dengan nama variabel *rand*.
3. Baris 3-15 merupakan proses pembangkitan populasi awal sejumlah *popsiz* yang telah ditentukan. Kromosom berisi gen-gen yang melambangkan tujuan wisata. Jumlah gen dalam satu kromosom mewakili jumlah tujuan wisata. Setelah itu susunan gen akan diacak. Pengacakan susunan gen mempunyai syarat, yaitu setiap kromosom tidak akan mempunyai susunan gen yang sama dengan kromosom lainnya.
4. Baris 16-17 merupakan proses inialisasi nilai parameter yang melekat pada setiap kromosom, yaitu *sigma*. Nilai *sigma* telah diatur dengan batas bawah, yaitu 0 dan batas atas, yaitu 2.
5. Baris 18-19 merupakan proses inialisasi nilai *fitness* tiap kromosom. Nilai *fitness* di proses pembangkitan populasi awal diatur dengan nilai *default*.

5.4.2 Reproduksi

Proses reproduksi dalam penelitian ini menggunakan proses mutasi. Proses mutasi dilakukan menggunakan metode *Reciprocal Exchange Mutation*. Dalam penelitian ini, nilai *lambda* (λ) ditentukan sebesar 2, yang berarti setiap individu akan menghasilkan 2 *offspring*. Implementasi kode program dari proses reproduksi dapat dilihat pada Kode Program 5.2.

No.	Kode Program
1	public void reproduksi() {
2	int indeksOffspring = 0;
3	Random rand = new Random();
4	for (int i = 0; i < this.p.length; i++) {
5	System.out.println("Individu ke "+i+" ");
6	for (int j = 0; j < this.offspring; j++) {
7	System.out.println("Mutasi ke "+j);
8	int prosesMutasi = (int)
9	Math.round(this.p[i][this.jmlDestinasi]);
10	if (prosesMutasi == 0) {
11	prosesMutasi = 1;
12	} this.c[indeksOffspring] =
13	Arrays.copyOfRange(this.p[i], 0, this.p[i].length);
14	for (int k = 0; k < prosesMutasi; k++) {
15	int titikTukarl = 0 + (int)
16	(Math.random() * ((this.jmlDestinasi - 1) - 0) + 1));
17	ArrayList<Integer> pengecualianRandom =
18	new ArrayList<>();
19	pengecualianRandom.add(titikTukarl);
20	int titikTukar2 =
21	randomDenganPengecualian(rand, 0, (this.jmlDestinasi - 1),
22	pengecualianRandom);

```

18 System.out.println("Swap      "+titikTukar1+      "      dengan
19      "+titikTukar2);
20      for (int l = 0; l < this.jmlDestinasi+2;
21      l++)
22      {
23      System.out.print(this.c[indeksOffspring][l]+" ");
24      }
25      System.out.println();
26
27      swap(this.c[indeksOffspring],
28      titikTukar1, titikTukar2);
29      for (int l = 0; l < this.jmlDestinasi+2;
30      l++) {
31      System.out.print(this.c[indeksOffspring][l]+" ");
32      }
33      System.out.println();
34      }
35      indeksOffspring++;
36      }
37      System.out.println("");
38      }
39      }
40      }
41      }
42      }
43      }
44      }
45      }
46      }
47      }
48      }
49      }
50      }
51      }
52      }
53      }
54      }
55      }
56      }
57      }
58      }
59      }
60      }
61      }
62      }
63      }
64      }
65      }
66      }
67      }
68      }
69      }
70      }
71      }
72      }
73      }
74      }
75      }
76      }
77      }
78      }
79      }
80      }
81      }
82      }
83      }
84      }
85      }
86      }
87      }
88      }
89      }
90      }
91      }
92      }
93      }
94      }
95      }
96      }
97      }
98      }
99      }
100     }
101     }
102     }
103     }
104     }
105     }
106     }
107     }
108     }
109     }
110     }
111     }
112     }
113     }
114     }
115     }
116     }
117     }
118     }
119     }
120     }
121     }
122     }
123     }
124     }
125     }
126     }
127     }
128     }
129     }
130     }
131     }
132     }
133     }
134     }
135     }
136     }
137     }
138     }
139     }
140     }
141     }
142     }
143     }
144     }
145     }
146     }
147     }
148     }
149     }
150     }
151     }
152     }
153     }
154     }
155     }
156     }
157     }
158     }
159     }
160     }
161     }
162     }
163     }
164     }
165     }
166     }
167     }
168     }
169     }
170     }
171     }
172     }
173     }
174     }
175     }
176     }
177     }
178     }
179     }
180     }
181     }
182     }
183     }
184     }
185     }
186     }
187     }
188     }
189     }
190     }
191     }
192     }
193     }
194     }
195     }
196     }
197     }
198     }
199     }
200     }
201     }
202     }
203     }
204     }
205     }
206     }
207     }
208     }
209     }
210     }
211     }
212     }
213     }
214     }
215     }
216     }
217     }
218     }
219     }
220     }
221     }
222     }
223     }
224     }
225     }
226     }
227     }
228     }
229     }
230     }
231     }
232     }
233     }
234     }
235     }
236     }
237     }
238     }
239     }
240     }
241     }
242     }
243     }
244     }
245     }
246     }
247     }
248     }
249     }
250     }
251     }
252     }
253     }
254     }
255     }
256     }
257     }
258     }
259     }
260     }
261     }
262     }
263     }
264     }
265     }
266     }
267     }
268     }
269     }
270     }
271     }
272     }
273     }
274     }
275     }
276     }
277     }
278     }
279     }
280     }
281     }
282     }
283     }
284     }
285     }
286     }
287     }
288     }
289     }
290     }
291     }
292     }
293     }
294     }
295     }
296     }
297     }
298     }
299     }
300     }
301     }
302     }
303     }
304     }
305     }
306     }
307     }
308     }
309     }
310     }
311     }
312     }
313     }
314     }
315     }
316     }
317     }
318     }
319     }
320     }
321     }
322     }
323     }
324     }
325     }
326     }
327     }
328     }
329     }
330     }
331     }
332     }
333     }
334     }
335     }
336     }
337     }
338     }
339     }
340     }
341     }
342     }
343     }
344     }
345     }
346     }
347     }
348     }
349     }
350     }
351     }
352     }
353     }
354     }
355     }
356     }
357     }
358     }
359     }
360     }
361     }
362     }
363     }
364     }
365     }
366     }
367     }
368     }
369     }
370     }
371     }
372     }
373     }
374     }
375     }
376     }
377     }
378     }
379     }
380     }
381     }
382     }
383     }
384     }
385     }
386     }
387     }
388     }
389     }
390     }
391     }
392     }
393     }
394     }
395     }
396     }
397     }
398     }
399     }
400     }
401     }
402     }
403     }
404     }
405     }
406     }
407     }
408     }
409     }
410     }
411     }
412     }
413     }
414     }
415     }
416     }
417     }
418     }
419     }
420     }
421     }
422     }
423     }
424     }
425     }
426     }
427     }
428     }
429     }
430     }
431     }
432     }
433     }
434     }
435     }
436     }
437     }
438     }
439     }
440     }
441     }
442     }
443     }
444     }
445     }
446     }
447     }
448     }
449     }
450     }
451     }
452     }
453     }
454     }
455     }
456     }
457     }
458     }
459     }
460     }
461     }
462     }
463     }
464     }
465     }
466     }
467     }
468     }
469     }
470     }
471     }
472     }
473     }
474     }
475     }
476     }
477     }
478     }
479     }
480     }
481     }
482     }
483     }
484     }
485     }
486     }
487     }
488     }
489     }
490     }
491     }
492     }
493     }
494     }
495     }
496     }
497     }
498     }
499     }
500     }
501     }
502     }
503     }
504     }
505     }
506     }
507     }
508     }
509     }
510     }
511     }
512     }
513     }
514     }
515     }
516     }
517     }
518     }
519     }
520     }
521     }
522     }
523     }
524     }
525     }
526     }
527     }
528     }
529     }
530     }
531     }
532     }
533     }
534     }
535     }
536     }
537     }
538     }
539     }
540     }
541     }
542     }
543     }
544     }
545     }
546     }
547     }
548     }
549     }
550     }
551     }
552     }
553     }
554     }
555     }
556     }
557     }
558     }
559     }
560     }
561     }
562     }
563     }
564     }
565     }
566     }
567     }
568     }
569     }
570     }
571     }
572     }
573     }
574     }
575     }
576     }
577     }
578     }
579     }
580     }
581     }
582     }
583     }
584     }
585     }
586     }
587     }
588     }
589     }
590     }
591     }
592     }
593     }
594     }
595     }
596     }
597     }
598     }
599     }
600     }
601     }
602     }
603     }
604     }
605     }
606     }
607     }
608     }
609     }
610     }
611     }
612     }
613     }
614     }
615     }
616     }
617     }
618     }
619     }
620     }
621     }
622     }
623     }
624     }
625     }
626     }
627     }
628     }
629     }
630     }
631     }
632     }
633     }
634     }
635     }
636     }
637     }
638     }
639     }
640     }
641     }
642     }
643     }
644     }
645     }
646     }
647     }
648     }
649     }
650     }
651     }
652     }
653     }
654     }
655     }
656     }
657     }
658     }
659     }
660     }
661     }
662     }
663     }
664     }
665     }
666     }
667     }
668     }
669     }
670     }
671     }
672     }
673     }
674     }
675     }
676     }
677     }
678     }
679     }
680     }
681     }
682     }
683     }
684     }
685     }
686     }
687     }
688     }
689     }
690     }
691     }
692     }
693     }
694     }
695     }
696     }
697     }
698     }
699     }
700     }
701     }
702     }
703     }
704     }
705     }
706     }
707     }
708     }
709     }
710     }
711     }
712     }
713     }
714     }
715     }
716     }
717     }
718     }
719     }
720     }
721     }
722     }
723     }
724     }
725     }
726     }
727     }
728     }
729     }
730     }
731     }
732     }
733     }
734     }
735     }
736     }
737     }
738     }
739     }
740     }
741     }
742     }
743     }
744     }
745     }
746     }
747     }
748     }
749     }
750     }
751     }
752     }
753     }
754     }
755     }
756     }
757     }
758     }
759     }
760     }
761     }
762     }
763     }
764     }
765     }
766     }
767     }
768     }
769     }
770     }
771     }
772     }
773     }
774     }
775     }
776     }
777     }
778     }
779     }
780     }
781     }
782     }
783     }
784     }
785     }
786     }
787     }
788     }
789     }
790     }
791     }
792     }
793     }
794     }
795     }
796     }
797     }
798     }
799     }
800     }
801     }
802     }
803     }
804     }
805     }
806     }
807     }
808     }
809     }
810     }
811     }
812     }
813     }
814     }
815     }
816     }
817     }
818     }
819     }
820     }
821     }
822     }
823     }
824     }
825     }
826     }
827     }
828     }
829     }
830     }
831     }
832     }
833     }
834     }
835     }
836     }
837     }
838     }
839     }
840     }
841     }
842     }
843     }
844     }
845     }
846     }
847     }
848     }
849     }
850     }
851     }
852     }
853     }
854     }
855     }
856     }
857     }
858     }
859     }
860     }
861     }
862     }
863     }
864     }
865     }
866     }
867     }
868     }
869     }
870     }
871     }
872     }
873     }
874     }
875     }
876     }
877     }
878     }
879     }
880     }
881     }
882     }
883     }
884     }
885     }
886     }
887     }
888     }
889     }
890     }
891     }
892     }
893     }
894     }
895     }
896     }
897     }
898     }
899     }
900     }
901     }
902     }
903     }
904     }
905     }
906     }
907     }
908     }
909     }
910     }
911     }
912     }
913     }
914     }
915     }
916     }
917     }
918     }
919     }
920     }
921     }
922     }
923     }
924     }
925     }
926     }
927     }
928     }
929     }
930     }
931     }
932     }
933     }
934     }
935     }
936     }
937     }
938     }
939     }
940     }
941     }
942     }
943     }
944     }
945     }
946     }
947     }
948     }
949     }
950     }
951     }
952     }
953     }
954     }
955     }
956     }
957     }
958     }
959     }
960     }
961     }
962     }
963     }
964     }
965     }
966     }
967     }
968     }
969     }
970     }
971     }
972     }
973     }
974     }
975     }
976     }
977     }
978     }
979     }
980     }
981     }
982     }
983     }
984     }
985     }
986     }
987     }
988     }
989     }
990     }
991     }
992     }
993     }
994     }
995     }
996     }
997     }
998     }
999     }
1000    }

```

Kode Program 5.2 Implementasi Reproduksi

Penjelasan Kode Program 5.2 yang merupakan implementasi reproduksi adalah sebagai berikut:

1. Baris 1 merupakan inisialisasi *method* bernama reproduksi().
2. Baris 2-30 merupakan proses mutasi yang dilakukan pada setiap kromosom di dalam *popsiz*e. Proses mutasi bergantung pada nilai *sigma* yang melekat pada setiap kromosom. Nilai *sigma* melambangkan jumlah proses mutasi yang harus dilakukan setiap individu untuk menghasilkan satu *offspring*. Pada penelitian ini setiap individu diatur untuk menghasilkan dua *offspring*.
3. Baris 35-37 merupakan inisialisasi *method* yang berisi inisialisasi variabel yang digunakan untuk menukarkan titik-titik dalam kromosom untuk proses mutasi.

5.4.3 Evaluasi

Setelah proses reproduksi dilakukan, maka selanjutnya adalah proses menggabungkan individu dalam populasi awal dan *offspring* untuk dihitung nilai *fitness*-nya. Implementasi kode program proses evaluasi dapat dilihat pada Kode Program 5.3.

No.	Kode Program
1	public void evaluasi() {
2	System.arraycopy(this.p, 0, this.pEval, 0,
3	this.popSize);
4	System.arraycopy(this.c, 0, this.pEval,
5	this.popSize, this.c.length);
6	for (int i = 0; i < this.pEval.length; i++) {
7	double fitness = hitungFitness(this.pEval[i]);
8	this.pEval[i][this.jmlDestinasi + 1] = fitness;
9	this.pEval[i][this.pEval[i].length - 1] =
10	fitness;
11	}
12	int indeks = 0;
13	int indeks2 = this.popSize;
14	int batas = 1;
15	for (int i = 0; i < this.c.length; i++) {
16	double sigma = 0;
17	if (this.c[i][this.jmlDestinasi + 1] >=
18	this.p[indeks][this.jmlDestinasi + 1]) {
19	sigma
20	this.pEval[indeks2][this.jmlDestinasi] * 1.1;
21	this.pEval[indeks2][this.jmlDestinasi]
22	sigma;
23	} else {
24	sigma
25	this.pEval[indeks2][this.jmlDestinasi] * 0.8;
26	this.pEval[indeks2][this.jmlDestinasi]
27	sigma;
28	}
29	if (batas == this.offspring) {
30	indeks++;
31	batas = 0;
32	}
33	batas++;
34	indeks2++;
35	}
36	}
37	public double hitungFitness(double arr[]) {
38	String [][]jamWisata = jamWisata(arr);
39	int totalJarak = 0;
40	for (int i = 0; i < this.jmlDestinasi - 1; i++) {
41	totalJarak
42	+= this.dataJarak[(int)
43	arr[i]][(int) arr[i + 1]];
44	}
45	int penalti = 0;
46	for (int i = 0; i < this.jmlDestinasi; i++) {
47	int selisih = Integer.parseInt(jamWisata[i][0])
48	-
49	Integer.parseInt(this.dataJamBuka[(int) arr[i]][0]);
50	if (selisih < 0) {
51	penalti += 100;
52	}
53	}
54	int
55	selisih2
56	=
57	Integer.parseInt(jamWisata[i][0])

No	Kode Program
	-
43	Integer.parseInt(this.dataJamBuka[(int) arr[i]][1]);
44	if (selisih2 >= 0) {
45	penalti += 85;
46	}
	int selisih3 = Integer.parseInt(jamWisata[i][1])
	-
47	Integer.parseInt(this.dataJamBuka[(int) arr[i]][1]);
48	if (selisih3 >= 0) {
49	penalti += 50;
50	}
51	return 1.0 / (totalJarak + (this.WAKTU_KUNJUNGAN * this.jmlDestinasi) + penalti);
52	}

Kode Program 5.3 Implementasi Evaluasi

Penjelasan Kode Program 5.3 yang merupakan implementasi evaluasi adalah sebagai berikut:

1. Baris 1 merupakan inisialisasi *method* bernama evaluasi().
2. Baris 2 merupakan inisialisasi variabel yang berguna sebagai penyimpanan *array* yang berisi individu dari populasi awal.
3. Baris 3 merupakan inisialisasi variabel yang digunakan untuk menyimpan *array* yang berisi individu dari populasi awal dan individu *offspring*.
4. Baris 4-8 merupakan proses perhitungan nilai *fitness* dari keseluruhan individu.
5. Baris 9-19 merupakan proses perhitungan nilai *sigma* tiap individu. Apabila individu mengalami kenaikan nilai *fitness*, maka nilai *sigma* akan dinaikkan dengan dikalikan 1,1. Apabila individu tidak mengalami kenaikan nilai *fitness*, maka nilai *sigma* akan diturunkan dengan dikalikan 0,8.
6. Baris 20-29 merupakan proses untuk *reset* indeks *sigma*.
7. Baris 30 merupakan inisialisasi *method* bernama hitungFitness().
8. Baris 31 merupakan proses mengambil jam wisata sementara kemudian disimpan ke dalam *array*.
9. Baris 32-35 merupakan proses menghitung total jarak tujuan wisata.
10. Baris 36-50 merupakan proses menghitung nilai penalti. Di dalam proses perhitungan nilai penalti, terdapat proses pengecekan antara jam buka maupun jam tutup wisata dengan jam mulai maupun jam selesai wisata. Dari hasil pengecekan tersebut, dapat diketahui besarnya nilai penalti tergantung dari tiap-tiap kondisi.
11. Baris 51-52 merupakan fungsi *return* terhadap rumus *fitness*.

5.4.4 Seleksi

Proses selanjutnya adalah melakukan seleksi untuk memilih individu yang pada akhirnya berhasil lolos ke generasi selanjutnya. Pada penelitian ini, metode proses seleksi yang digunakan adalah metode *elitism selection*. Implementasi kode program proses seleksi dapat dilihat pada Kode Program 5.4.

No.	Kode Program
1	public void seleksi() {
2	double tempSeleksi[][] = new
	double[this.pEval.length][this.pEval[0].length];
3	double hasilSeleksi[][] = new
	double[this.popSize][this.pEval[0].length];
4	System.arraycopy(this.pEval, 0, tempSeleksi, 0,
	this.pEval.length);
5	tempSeleksi = elitism(tempSeleksi);
6	System.arraycopy(tempSeleksi, 0, hasilSeleksi, 0,
	this.popSize);
7	updatePopulasi(hasilSeleksi);
8	}
9	public double[][] elitism(double[][] pSeleksi) {
10	double[][] tempP = pSeleksi;
11	Arrays.sort(tempP, new
	java.util.Comparator<double[]>() {
	public int compare(double[] a, double[] b) {
	return Double.compare(b[b.length - 1],
	a[a.length - 1]);
	}
12	});
13	return tempP;
14	}
15	public void updatePopulasi(double pSeleksi[][]) {
16	this.p = pSeleksi;
17	}

Kode Program 5.4 Implementasi Seleksi

Penjelasan Kode Program 5.4 yang merupakan implementasi seleksi adalah sebagai berikut:

1. Baris 1 merupakan inialisasi *method* bernama seleksi().
2. Baris 2 merupakan inialisasi *array* bernama tempSeleksi[][] yang digunakan untuk penyimpanan sementara proses seleksi.
3. Baris 3 merupakan inialisasi *array* bernama hasilSeleksi[][] yang digunakan untuk penyimpanan hasil akhir proses seleksi.
4. Baris 4-8 merupakan proses seleksi yang memanggil *method* bernama elitism().
5. Baris 9-14 merupakan proses yang terjadi di dalam metode *elistism selection*.
6. Baris 15-17 inialisasi *method* bernama updatePopulasi() yang berisi proses memperbarui populasi yang lolos ke generasi selanjutnya dengan hasil dari proses seleksi.

5.5 Implementasi Antarmuka

Antarmuka merupakan jembatan penghubung antara pengguna dan sistem. Pengguna dapat memberikan masukan pada sistem melalui antarmuka. Antarmuka juga dapat menampilkan keluaran dari hasil perhitungan *Evolution Strategies* pada pengguna. Implementasi antarmuka dapat dilihat pada Gambar 5.1.

Selamat Datang di Aplikasi Rekomendasi Wisata Kota Batu

Silahkan pilih tujuan wisata

☒ Jawa Timur Park 1 ☒ Museum Angkut ☐ Wisata Paralayang Gunung Banyak

☐ Jawa Timur Park 2 ☐ Predator Fun Park ☐ Kaliwatu Rafting

☐ Jawa Timur Park 3 ☒ Selecta ☐ Coban Rais

☒ Batu Secret Zoo ☐ Batu Wonderland ☐ Coban Talun

☐ Eco Green Park ☐ Kusuma Agrowisata ☐ Coban Putri

☒ Batu Night Spectacular ☐ Alun-Alun Kota Wisata ...

Silahkan masukan waktu wisata

Jam mulai: 08 30 Jam selesai: 02 00

Generasi: 1

SUBMIT

OUTPUT

Jam Mulai	Jam Selesai	Lokasi
08:30	10:30	Jawa Timur Park 1
10:36	12:36	Batu Secret Zoo
13:00	15:00	Selecta
15:20	17:20	Museum Angkut
17:28	19:28	Batu Night Spectacular

Gambar 5.1 Implementasi Antarmuka

Penjelasan antarmuka pada Gambar 5.1 adalah sebagai berikut:

1. Antarmuka bagian atas adalah bagian yang berguna sebagai masukan untuk pengguna. Ada 17 pilihan tujuan wisata yang dapat dipilih oleh pengguna. Untuk mengatur waktu wisata, pengguna dapat mengatur jam mulai dan jam selesai wisata sesuai keinginan. Kemudian pengguna juga dapat memasukkan jumlah generasi yang dikehendaki untuk proses perhitungan *Evolution Strategies*.
2. Antarmuka bagian bawah merupakan bagian untuk tampilan keluaran dari hasil perhitungan *Evolution Strategies*.

BAB 6 PENGUJIAN DAN ANALISIS

Hal yang akan dibahas pada bab pengujian dan analisis ini terkait proses pengujian Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies*. Pengujian yang dilakukan adalah pengujian parameter. Terdapat dua jenis parameter, yaitu parameter utama dan parameter pendukung. Parameter utama meliputi ukuran populasi, jumlah generasi, dan jumlah *offspring*. Sedangkan parameter pendukung meliputi tujuan wisata, waktu berangkat wisata, dan waktu selesai wisata. Tujuan utama dari tahap pengujian pada penelitian ini adalah untuk mengetahui nilai parameter utama yang optimal, sehingga dapat menghasilkan rute yang efisien. Berikut parameter-parameter yang terdapat pada sistem dapat dilihat pada Tabel 6.1.

Tabel 6.1 Parameter Pengujian

No	Nama Parameter	Jenis Parameter	Nilai Parameter
1.	Ukuran Populasi	Utama	Mengikuti pengujian
2.	Jumlah Offspring	Utama	Mengikuti pengujian
3.	Jumlah Generasi	Utama	Mengikuti pengujian
4.	Tujuan Wisata	Pendukung	Jawa Timur Park 1, Jawa Timur Park 2, Batu Secret Zoo, Batu Night Spectacular, Museum Angkut, Selecta
5.	Waktu Mulai Wisata	Pendukung	08.30
6.	Waktu Selesai Wisata	Pendukung	22.00

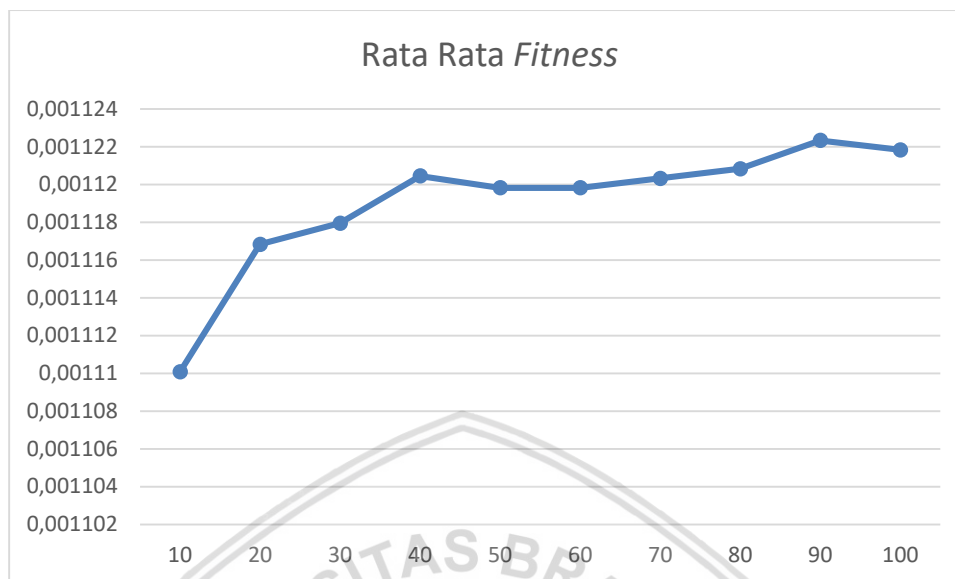
6.1 Pengujian Ukuran Populasi

Mengganti nilai ukuran populasi merupakan cara menguji ukuran populasi berapa yang menghasilkan nilai *fitness* paling besar. Nilai yang paling awal diuji dari ukuran populasi yaitu sebanyak 10, kemudian dilanjutkan dengan kelipatan 10 hingga ukuran populasi 100. Setiap pengujian untuk satu nilai ukuran populasi dilakukan sebanyak 10 kali. Dalam pengujian ini, jumlah generasi adalah 3 dan jumlah *offspring* adalah 3. Hasil pengujian ukuran populasi dapat dilihat pada Tabel 6.2.

Tabel 6.2 Hasil Pengujian Ukuran Populasi

Ukuran Populasi	Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
	Percobaan										
	1	2	3	4	5	6	7	8	9	10	
10	0,0010964	0,001116	0,0010964	0,0011173	0,0011086	0,0011098	0,001116	0,0011223	0,0011001	0,0011173	0,001110073
20	0,0011148	0,0011173	0,0011223	0,0011173	0,0011173	0,001116	0,0011173	0,0011223	0,001116	0,0011074	0,001116833
30	0,001116	0,0011223	0,0011173	0,0011148	0,0011135	0,0011223	0,0011223	0,0011173	0,0011173	0,001116	0,001117951
40	0,0011173	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,001116	0,0011148	0,001120456
50	0,0011223	0,0011173	0,0011223	0,0011173	0,0011223	0,0011173	0,0011173	0,0011223	0,0011223	0,0011173	0,001119826
60	0,0011173	0,0011173	0,0011173	0,0011223	0,0011173	0,0011173	0,0011223	0,0011223	0,0011223	0,0011223	0,001119826
70	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,0011173	0,0011173	0,0011223	0,0011173	0,0011223	0,001120328
80	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,0011173	0,0011173	0,0011223	0,0011223	0,0011223	0,00112083
90	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,001122334
100	0,0011223	0,0011223	0,0011223345	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,001121833

Didapatkan rata-rata nilai *fitness* berupa grafik dari hasil pengujian ukuran populasi berdasarkan Tabel 6.2 yang dapat dilihat pada Gambar 6.1.



Gambar 6.1 Grafik Nilai *Fitness* Pengujian Ukuran Populasi

Grafik nilai rata-rata *fitness* pengujian ukuran populasi seperti Gambar 6.1, menunjukkan bahwa nilai rata-rata *fitness* cenderung mengalami kenaikan. Kecuali pada pengujian ukuran populasi 50 yang mengalami penurunan dibanding ukuran populasi 40. Pada pengujian ukuran populasi 100 juga mengalami penurunan apabila dibandingkan dengan ukuran populasi 90. Namun secara keseluruhan nilai rata-rata *fitness* dapat dikatakan mengalami kenaikan.

Ketika ukuran populasi sebesar 90 populasi didapatkan nilai rata-rata *fitness* tertinggi dengan nilai rata-rata *fitness* 0,001122. Namun nilai tetap tergantung dari variasi masukan pilihan wisata oleh pengguna, karena panjang suatu kromosom merupakan representasi dari tujuan wisata.

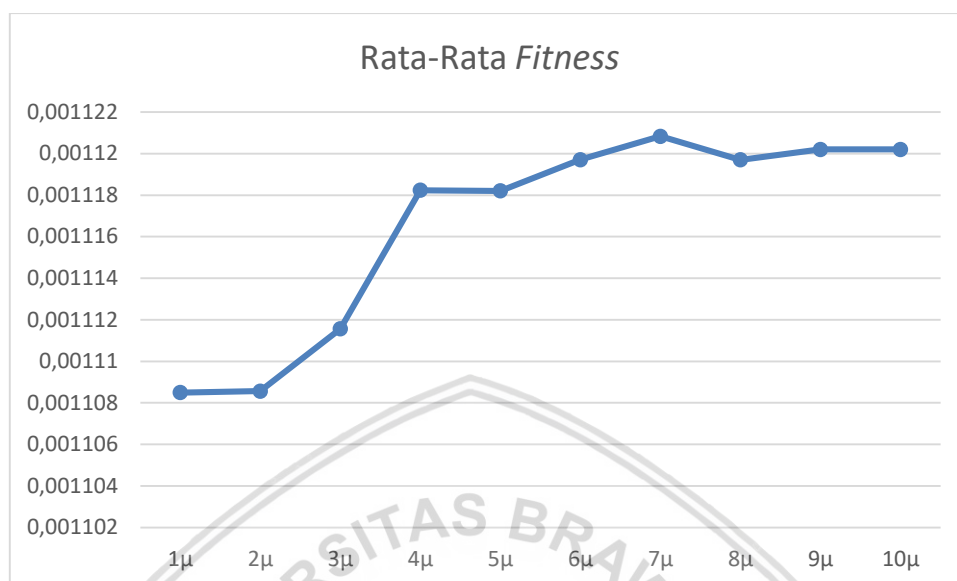
6.2 Pengujian Jumlah *Offspring*

Untuk mengetahui pengaruh nilai *fitness* dengan banyaknya *offspring* atau jumlah *child* yang dihasilkan pada saat proses reproduksi menggunakan metode mutasi, maka dilakukanlah pengujian jumlah *offspring*. Dengan dilakukan pengujian ini, diharapkan dapat diketahui jumlah *offspring* yang terbaik untuk digunakan dalam sistem. Pengujian jumlah *offspring* dimulai dari jumlah *offspring* sebesar 1 μ hingga 10 μ . Masing-masing *offspring* melakukan 10 kali pengujian untuk mencari nilai rata-rata *fitness* yang terbaik. Pengujian ini menggunakan ukuran populasi sebesar 10 dan jumlah generasi 3. Hasil pengujian jumlah *offspring* dapat dilihat pada Tabel 6.3.

Tabel 6.3 Hasil Pengujian Jumlah Offspring

Jumlah Offspring	Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
	Percobaan										
	1	2	3	4	5	6	7	8	9	10	
1μ	0,0011173	0,0011001	0,0011148	0,0011086	0,0010881	0,0011086	0,0011223	0,0011173	0,0010976	0,0011098	0,001108492
2μ	0,0010764	0,0011223	0,0011098	0,0011173	0,0011223	0,0011001	0,0011135	0,0011173	0,0011086	0,0010976	0,001108565
3μ	0,001094	0,0010964	0,0011173	0,0011223	0,0011098	0,001116	0,0011173	0,0011098	0,0011098	0,0011223	0,001111559
4μ	0,0011173	0,0011223	0,0011223	0,0011135	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,0011001	0,001118234
5μ	0,0011086	0,0011223	0,0011173	0,0011223	0,0011173	0,0011173	0,0011173	0,0011148	0,0011223	0,0011223	0,001118209
6μ	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,0011223	0,0011148	0,001116	0,0011223	0,0011148	0,001119705
7μ	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011173	0,0011173	0,0011223	0,0011173	0,0011223	0,00112083
8μ	0,0011223	0,001116	0,0011223	0,0011173	0,0011223	0,0011223	0,0011223	0,0011173	0,0011173	0,0011173	0,001119702
9μ	0,001116	0,0011173	0,0011223	0,0011223	0,0011173	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,001120203
10μ	0,0011173	0,001116	0,0011223	0,0011223	0,0011223	0,0011173	0,0011173	0,0011223	0,0011223	0,0011223	0,001120203

Didapatkan rata-rata nilai *fitness* berupa grafik yang diperoleh berdasarkan Tabel 6.3, maka hasil pengujian jumlah *offspring* yang dapat dilihat pada Gambar 6.2.



Gambar 6.2 Grafik Nilai *Fitness* Pengujian Jumlah *Offspring*

Berdasarkan pada grafik nilai rata-rata *fitness* pengujian jumlah *offspring* seperti Gambar 6.2, menunjukkan bahwa nilai rata-rata *fitness* cenderung mengalami kenaikan, kecuali pada saat ukuran *offspring* 8μ yang mengalami penurunan dibandingkan ukuran *offspring* 7μ.

Nilai rata-rata *fitness* yang terbaik didapatkan pada saat ukuran *offspring* sebesar 7μ, dengan nilai rata-rata *fitness* 0,00112083. Nilai ini tetap tergantung dengan variasi masukan pilihan wisata yang dipilih oleh pengguna, karena panjang kromosom ditentukan oleh pilihan tujuan wisata.

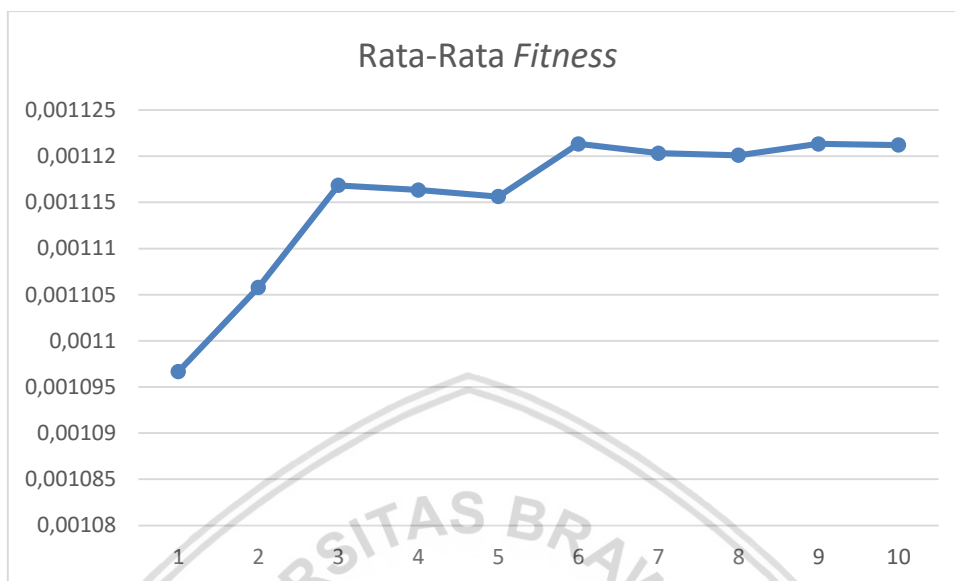
6.3 Pengujian Jumlah Generasi

Untuk mengetahui berapa nilai jumlah generasi yang dapat menghasilkan nilai *fitness* terbaik, maka dilakukanlah pengujian jumlah generasi. Cara menguji jumlah generasi yang menghasilkan nilai *fitness* paling besar adalah dengan dilakukan penggantian nilai jumlah generasi secara bertahap dimulai dari jumlah generasi sebesar 1 hingga 10. Masing-masing nilai jumlah generasi melakukan pengujian sebanyak 10 kali untuk mendapatkan nilai rata-rata *fitness* yang terbaik. Pengujian ini menggunakan ukuran populasi sebesar 10 dan jumlah *offspring* 3. Hasil akhir pengujian jumlah generasi dapat dilihat pada Tabel 6.4.

Tabel 6.4 Hasil Pengujian Jumlah Generasi

Jumlah Generasi	Nilai <i>Fitness</i>										Rata-Rata Fitness
	Percobaan										
	1	2	3	4	5	6	7	8	9	10	
1	0,0010976	0,0011135	0,0010683	0,0010845	0,0011135	0,0010881	0,0010683	0,0011001	0,0011148	0,0011173	0,001096661
2	0,0011001	0,0011223	0,0011049	0,0011001	0,0011173	0,0011049	0,0011049	0,0010928	0,001116	0,001094	0,001105785
3	0,0011173	0,0011223	0,0011173	0,0011173	0,0011061	0,0011223	0,0011173	0,0011223	0,0011173	0,0011086	0,001116844
4	0,0011173	0,0011173	0,0011173	0,0011173	0,0011173	0,0011223	0,0011173	0,0011098	0,0011049	0,0011223	0,001116343
5	0,0011223	0,0011223	0,0011223	0,0011148	0,0011173	0,0011173	0,0011173	0,0011223	0,0011001	0,0011001	0,001115634
6	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,0011173	0,0011223	0,0011223	0,0011223	0,0011223	0,001121331
7	0,0011173	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,0011223	0,0011223	0,0011173	0,0011173	0,001120328
8	0,0011223	0,0011049	0,0011223	0,0011173	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,001120097
9	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,0011173	0,0011223	0,0011173	0,0011223	0,001121331
10	0,0011223	0,0011223	0,0011223	0,001116	0,0011173	0,0011223	0,0011223	0,0011223	0,0011223	0,0011223	0,001121207

Didapatkan rata-rata nilai *fitness* berupa grafik yang diperoleh berdasarkan Tabel 6.4, maka hasil dari pengujian jumlah generasi yang dapat dilihat pada Gambar 6.3.



Gambar 6.3 Grafik Nilai *Fitness* Pengujian Jumlah Generasi

Nilai rata-rata *fitness* pengujian jumlah generasi dapat ditunjukkan berupa grafik seperti Gambar 6.3, diketahui bahwa nilai rata-rata *fitness* cenderung mengalami kenaikan, kecuali pada saat jumlah generasi sebesar 7 yang mengalami penurunan dibandingkan jumlah generasi sebesar 6.

Pada saat generasi sebesar 6 diperoleh nilai rerata *fitness* terbaik, dengan nilai rata-rata *fitness* 0,001121331. Nilai ini tetap tergantung dengan variasi masukan pilihan wisata yang dipilih oleh pengguna, karena panjang kromosom ditentukan oleh pilihan tujuan wisata.

6.4 Pengujian Parameter Secara Global

Pengujian ini dilakukan untuk mengetahui nilai *fitness* tertinggi dengan menggunakan parameter terbaik yang dihasilkan selama pengujian sebelumnya. Parameter yang digunakan meliputi, ukuran populasi sebanyak 90, jumlah *offspring* sebanyak 7 μ , dan jumlah generasi sebanyak 6. Hasil pengujian dapat dilihat pada Gambar 6.4.

```

Output - skripsi (run)
Nilai Fitness :
0.0013071895
Hasil Rekomendasi Tujuan Wisata
Jam Mulai      Jam Selesai      Lokasi
08:30          10:30           Selecta
10:55          12:55           Jawa Timur Park 1
13:01          15:01           Jawa Timur Park 2
15:02          17:02           Batu Secret Zoo
17:07          19:07           Museum Angkut
19:15          21:15           Batu Night Spectacular
BUILD SUCCESSFUL (total time: 1 second)
    
```

Gambar 6.4 Hasil Pengujian Global

BAB 7 PENUTUP

Penelitian ini akan ditutup dengan membahas kesimpulan yang diperoleh dari hasil pengujian yang telah dilakukan dalam penelitian ini. Selain itu akan disertakan juga saran untuk keperluan dalam pengembangan atau kelanjutan dari penelitian ini.

7.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis yang telah dilakukan sebelumnya, maka dapat disimpulkan mengenai Optimasi *Travelling Salesman Problem With Time Windows* Pada Sistem Rekomendasi Tujuan Wisata Di Kota Batu Dengan Metode *Evolution Strategies* adalah sebagai berikut:

1. *Evolution Strategies* (ES) dapat digunakan sebagai metode penyelesaian permasalahan dalam pencarian rute perjalanan wisata terbaik di Kota Batu. Penelitian ini dilengkapi dengan parameter utama yang meliputi ukuran populasi, jumlah *offspring*, dan jumlah generasi serta parameter pendukung yang meliputi tujuan wisata, waktu mulai wisata, dan waktu selesai wisata.
2. Pada penelitian ini terdapat dua parameter, yaitu parameter utama dan parameter pendukung. Parameter yang berpengaruh secara langsung pada kinerja algoritme *Evolution Strategies* (ES) yang disebut parameter utama. Ukuran populasi, jumlah *offspring*, dan jumlah generasi merupakan parameter utama. Parameter yang nilainya dapat disesuaikan dengan keinginan pengguna disebut dengan parameter pendukung. Tujuan wisata, waktu mulai wisata, dan waktu selesai wisata masuk ke dalam kategori parameter pendukung.
3. Berdasarkan pengujian, parameter-parameter tersebut berpengaruh pada nilai *fitness* yang dihasilkan, terutama parameter utama. Nilai *fitness* yang diperoleh, sangat dipengaruhi oleh ukuran populasi. Semakin besar ukuran populasi, maka semakin beragam susunan kromosomnya. Pada pengujian ukuran populasi, nilai rata-rata *fitness* terbaik dihasilkan pada saat ukuran populasi sebesar 90 dengan nilai 0,001122. Jumlah *offspring* berpengaruh terhadap variasi kromosom hasil proses reproduksi dan secara tidak langsung juga mempengaruhi susunan populasi baru. Berdasarkan hasil pengujian jumlah *offspring*, didapatkan nilai rata-rata *fitness* terbaik pada saat jumlah *offspring* sebesar 7 μ , dengan nilai rata-rata *fitness* 0,00112083. Sedangkan pengaruh jumlah generasi tidak terlalu terlihat secara signifikan hasil perubahannya mulai dari generasi ke 6 hingga generasi ke 10. Pada saat generasi ke 6 diperoleh rata-rata *fitness* terbaik, dengan nilai 0,001121331. Mudah diperoleh kesimpulan bahwa pada generasi ke 6 telah muncul solusi terbaik dari penyelesaian masalah menggunakan metode *Evolution Strategies* (ES). Hasil pengujian tetap tergantung dari jumlah pilihan tujuan wisata yang dimasukkan oleh pengguna. Hal ini dikarenakan susunan gen dalam setiap

kromosom yang terbentuk merupakan representasi pilihan tujuan wisata yang dipilih oleh pengguna.

7.2 Saran

Sejauh penelitian ini telah berlangsung, dapat diperoleh saran untuk dapat melakukan pengembangan lebih lanjut tentang topik yang serupa adalah sebagai berikut:

1. Sistem dapat dibuat menggunakan metode *Evolution Strategies* yang di-*hybrid* dengan metode optimasi lainnya seperti *Ant Colony Optimization*, *Particle Swarm Optimization*, ataupun *Artificial Bee Colony*. Dengan asumsi, apabila metode *Evolution Strategies* di-*hybrid* dengan metode yang tepat, maka akan menghasilkan rute perjalanan wisata yang lebih optimal.
2. Sistem dapat dikembangkan ke dalam bentuk *mobile app*, karena lebih praktis apabila digunakan ketika berpergian.



DAFTAR PUSTAKA

- Vista, C. B., 2015. *Penerapan Algoritma Evolution Strategies Untuk Optimasi Distribusi Barang Dua Tahap*. S1. Universitas Brawijaya Malang. *SKRIPSI*.
- Endarwati, D. A., 2014. Pencarian Rute Optimum Dengan Evolution Strategies. *SKRIPSI*.
- Harun, I. A., 2014. Implementasi Evolution Strategies untuk Penyelesaian Vehicle Routing Problem With Time Windows pada Distribusi Minuman Soda XYZ. *SKRIPSI*.
- Mahmudy, W. F., 2013. Algoritma Evolusi. *Modul Kuliah Program Teknologi Informasi dan Ilmu Komputer (PTI IK) Universitas Brawijaya Malang*.
- Munawaroh, F., 2015. *Optimasi Distribusi Pupuk Menggunakan Evolution Strategies*. S1. Universitas Brawijaya Malang. *SKRIPSI*.
- Nissa, C., 2013. Optimasi Rute Angkutan Kota Malang dengan Penerapan Algoritma Genetika. *SKRIPSI*.
- Priandani, N. D., 2015. Optimasi Travelling Salesman Problem With Time Windows (TSP-TW) Pada Penjadwalan Paket Rute Wisata di Pulau Bali Menggunakan Algoritma Genetika. *SKRIPSI*.
- Purwana, N., 2016. Optimalisasi Penempatan Dosen Pembimbing dan Penjadwalan Seminar Tugas Akhir Menggunakan Algoritma Genetika. *SKRIPSI*.
- Widodo, A. W., 2017. Penerapan Evolution Strategies Untuk Optimasi Travelling Salesman Problem With Time Windows Pada Sistem Rekomendasi Wisata Malang Raya. *Jurnal Ilmiah KURSOR*.
- Widodo, A. W. & Mahmudy, W. F., 2010. Penerapan Algoritma Genetika pada Sistem Rekomendasi Wisata Kuliner. *Jurnal Ilmiah KURSOR*.

LAMPIRAN

A.1 Data Tempat Wisata

No	Nama Lokasi	Waktu	
		Buka	Tutup
1	Jawa Timur Park 1	08.30	16.30
2	Jawa Timur Park 2	09.00	18.00
3	Jawa Timur Park 3	11.00	18.00
4	Batu Secret Zoo	09.30	17.30
5	Eco Green Park	09.00	17.00
6	Batu Night Spectacular	15.00	00.00
7	Museum Angkut	12.00	20.00
8	Predator Fun Park	08.30	16.30
9	Selecta	06.00	17.00
10	Batu Wonderland	09.00	17.00
11	Kusuma Agrowisata	07.00	17.00
12	Alun-Alun Kota Wisata Batu	00.00	23.59
13	Wisata Paralayang Gunung Banyak	00.00	23.59
14	Kaliwatu Rafting	08.00	17.00
15	Coban Rais	00.00	16.00
16	Coban Talun	00.00	17.00
17	Coban Putri	00.00	16.00

A.2 Data Waktu Tempuh

ID Tempat Wisata	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	0	6	11	6	5	7	4	16	25	5	7	8	32	12	19	30	15
2	6	0	12	1	3	6	7	16	26	6	10	11	35	14	17	31	13
3	11	12	0	10	11	9	11	8	27	7	14	10	36	15	20	32	13
4	6	1	10	0	1	4	5	13	24	4	8	7	33	13	16	30	12
5	5	3	11	1	0	6	7	15	26	6	10	9	35	15	17	31	13
6	7	6	9	4	6	0	8	9	28	7	11	11	36	16	14	33	9
7	4	7	11	5	7	8	0	17	20	5	3	5	28	8	16	26	12
8	16	16	8	13	15	9	17	0	37	16	21	20	45	25	13	42	7
9	25	26	27	24	26	28	20	37	0	24	26	18	35	17	40	15	36
10	5	6	7	4	6	7	5	16	24	0	8	5	31	10	19	27	15
11	7	10	14	8	10	11	3	21	26	8	0	9	31	12	14	30	10
12	8	11	10	7	9	11	5	20	18	5	9	0	26	5	22	23	17
13	32	35	36	33	35	36	28	45	35	31	31	26	0	26	41	44	37
14	12	14	15	13	15	16	8	25	17	10	12	5	26	0	27	22	23
15	19	17	20	16	17	14	16	13	40	19	14	22	41	27	0	43	6
16	30	31	32	30	31	33	26	42	15	27	30	23	44	22	43	0	47
17	15	13	13	12	13	9	12	7	36	15	10	17	37	24	6	47	0